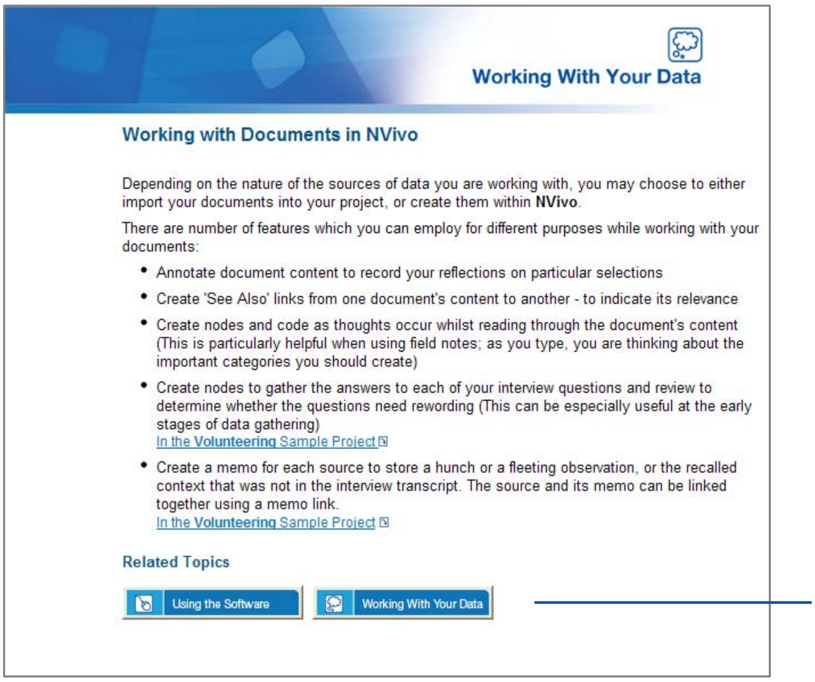
Working With Your Data

NVivo 8 Help

Working With Your Data

This is a printable version of the NVivo 8 help that is built into the software. The help is divided into two sections – ‘Using the Software’ and ‘Working with Your Data’. This section of the help provides conceptual guidance and strategies for handling your data in NVivo.



Working With Your Data

Working with Documents in NVivo

Depending on the nature of the sources of data you are working with, you may choose to either import your documents into your project, or create them within **NVivo**.

There are number of features which can you can employ for different purposes while working with your documents:

- Annotate document content to record your reflections on particular selections
- Create 'See Also' links from one document's content to another - to indicate its relevance
- Create nodes and code as thoughts occur whilst reading through the document's content (This is particularly helpful when using field notes; as you type, you are thinking about the important categories you should create)
- Create nodes to gather the answers to each of your interview questions and review to determine whether the questions need rewording (This can be especially useful at the early stages of data gathering)
  In the Volunteering Sample Project
- Create a memo for each source to store a hunch or a fleeting observation, or the recalled context that was not in the interview transcript. The source and its memo can be linked together using a memo link.
  In the Volunteering Sample Project

Related Topics

To electronically navigate through this help and link to other topics, go to the **NVivo Help** under the **Help** menu in your NVivo 8 software.

Click a button for related instructions or concepts.

Produced November 2008.

Copyright © 2008 QSR International Pty Ltd   ABN 47 006 357 213.

All rights reserved.

NVivo and QSR words and logos are trademarks or registered trademarks of QSR International Pty Ltd. Patent pending. Microsoft, .NET, SQL Server, Windows, XP, Vista, Windows Media Player, Word, PowerPoint and Excel are trademarks or registered trademarks of the Microsoft Corporation in the United States and/or other countries. Adobe, .pdf and Flash Player are trademarks or registered trademarks of Adobe Systems Integrated in the United States and/or other countries. QuickTime and the QuickTime logo are trademarks or registered trademarks of Apple Computer, Inc., used under license there from. Crystal Reports is a trademark or registered trademark of Business Objects SA. This information is subject to change without notice.

Contents

What is Qualitative Research?

Qualitative research is a term applied to a wide range of methods for handling materials that are relatively unstructured and not appropriately reduced to numbers. Working with transcripts of interviews, focus groups, or audio/video files or with field notes or other documents, researchers seek new understanding of a situation, experience or process. Goals vary from complete understanding of a process over time to swift discovery and illustration of themes.

These methods are used across social and health sciences, education, market and business research, program evaluation, legal, political and historical studies, life histories and policy assessment. Different methodologies require different ways of handling and interpreting the data.

Qualitative research requires both creativity and data management—sensitive interpretation of complex data, accurate access to information and ways of rigorously exploring themes and discovering and testing patterns. Software can assist in all these processes and **NVivo 8** is designed to integrate them.

Approaching an NVivo Project

Where to Begin?

A qualitative project may seem very open-ended, with messy data, no specific starting tasks to do, tests to perform or samples to acquire. In **NVivo**, there is no reason to delay starting. The software is designed to assist you in managing dissimilar sources of data and seemingly uncoordinated ideas.

Start a project as soon as you have any reflections on the topic (your correspondence about research design, grant application, thesis proposal or literature notes). These materials can be used to set up the project and to create early themes or topics for analysis.

What is a Project in NVivo?

To do research using **NVivo**, you create a project to hold all the data and ideas that you bring to the task or will generate as you work.

Your **NVivo** project is the container for everything you bring to the study of this research topic—your data, ideas and the connections between them.

When Should I Start a Project?

You can start your project in **NVivo** as soon as you start your research. There is no need to wait until you are confident of the shape of the project, since an **NVivo** project can change and grow flexibly. There is no advantage in waiting until "all the data are in". Qualitative projects often commence with a very limited design, reshaping as they are driven by discoveries in the data.

You do not need to have done any prior planning or design before starting a project in **NVivo**. But if you have designed your research you can set up sources, cases and nodes to support that design and **NVivo** folders to organize the material.

Securing Your Data

NVivo provides a number of features to help you secure your project:

'Undo' Functionality

> If you make a mistake when working with your project you can use **NVivo**'s 'Undo' function to undo (up to five) most recent actions. You can also use the undo function to try an action without keeping the results—this can be useful as qualitative research is by nature exploratory and creative.

Save Reminders

> **NVivo** optionally provides reminders for you to save at nominated intervals. It's wise to keep these reminders frequent when a project is in very active phase of growth: later it may be safe to make the interval longer. Qualitative projects grow rapidly and often in unpredictable ways, so it is important to save your project frequently.

Deletion Confirmation Messages

> **NVivo** optionally provides confirmation messages before you delete items in you project.

'Copy Project' Functionality

> It is useful to make frequent copies of your project, as you find that your analysis has altered your project considerably in a direction you now regret. Return to an earlier version to rework data or pursue an alternative analytical route.

Project Passwords

NVivo provides two types of passwords for your project, read only and read/write. If you want to prevent unauthorized access to your project, you can set the read/write password. Also, you can provide a copy of your project for someone to review but not change by setting a read-only password for it.

Always ensure you have a current backup of your project.

Working with Multiple Projects

At some point in your project you may need to merge your **NVivo** project with another. Your project may call for a number of team members working with different sources of data or perhaps with different approaches to analyzing the same sources.

Reasons for Merging

It is very important to be clear about your reasons for merging projects. These may be central to your approach to your research and data, such as:

- To bring together multiple bodies of data under the same analysis umbrella (i.e. data collected by different members of a team)
- To facilitate different analyses of the same data (i.e. from different members of a team with different analytical approaches)
- To compare the data and outcomes from an earlier project with another (i.e. for a longitudinal project)

On the other hand, you may wish to share the framework of your project (i.e. node hierarchies, sets, attributes and values), rather than its content. For instance, you may want to:

- Share patterns of analysis between researchers
- Commence a similar or new related project quickly
- Create a project structure (or template) for use by all team members

Planning

As you plan your research, consider the points at which projects should be merged. This could be after the completion of first round interviews or once topic coding has reached a mature stage. This will ensure that you are able to keep a big picture view of all data sources and analysis processes in your project, even though they are effectively separate.

Choosing Merge Outcomes

As you merge your projects, ensure your reasons for merging are reflected in the options you choose. For example,

- If your aim is to bring together different data from team members in different areas who interviewed different participants in different geographical areas, it is possible that some of these participants have the same name. As you would like to merge both projects' content, it is important to import all items, including their content. Also, as it is likely that there will be some duplication in your cases, it is important to choose not to merge duplicate items but to create them as new items.

- If a team member wants to share their project's tree node hierarchies (or other nodes) so that you can use it in your analysis, it is unlikely that you want to merge the content of their project with your own. Therefore, it is important to import only the tree nodes, excluding their content, into your project.

It is recommended that you back up your project before importing another.

Organizing Your Data

As your project grows you are likely to build up a large number of sources, nodes, queries and models. You may want to group these items in the following ways:

Using Subfolders

You can create subfolders for your sources, queries and models to assist in the organization of your project:

- You can use folders to organize sources based on their distinct types (e.g. interviews, focus groups and diaries). You could also distinguish further source types (e.g. stages in data gathering).
- Queries could be stored in folders relevant to their processes within your project (e.g. by analysis sector, or chapter, or overarching theme).
- Models may cover a wide range of analysis areas which can be reflected in folders

In the **Volunteering** Sample Project

The interview documents, audio and video are stored in a folder called **Interviews** and focus group documents and video are separately stored in a folder called **Focus Groups**.

A separate folder called **Project Notes** keeps documents related to the running of the project separate to it sources of data.

Using Sets

Sets are a swift and flexible way of grouping sources and/or nodes. When you place an item in a set **NVivo** creates an alias (or shortcut) to it. Hence there is no limit to the number of sets a source or node may be aliased to.

You might use sets in the following ways:

- To order and organize a set of pictures - put picture sources in a set and display them as thumbnails to make a 'photo gallery'.
- To see stages, progress and changes in data construction, or to manage research timetables (e.g. "Items created this week", "Nodes without coding")
- To direct and inform coding (e.g. "Sources not yet coded", "Nodes created since team debate")
- To manage coding and auto coding (e.g. "Nodes to discuss and re-describe", "Sources not yet auto coded")
- Group items around a theme or an area of analysis
- Identify and compare the work of team members, a set for each can contain the interviews they conducted, memos they wrote and nodes they introduced

In the **Volunteering** Sample Project

The set **Simon's Photo Diary** contains an ordered set of pictures taken by a research participant.

The set **Review Coding** contains sources that have been coded but need to be reviewed.

Meeting the Challenges of Teamwork

A growing number of projects involve multiple researchers working collaboratively. While teams offer higher productivity and a richer perspective, they also present a number of management challenges. Early in a project it is important to determine the approach your team will take to:

- Collecting and organizing data
- Creating and cataloguing themes and topics (the node structure)
- Coding the data

Collecting and Organizing Data

Many large projects - including those spread over time or geographical location - need to manage an ever-growing volume of data. **NVivo** can help in the following ways:

- Keep all the data in a central 'master' project—one that is ideally managed by a team leader or data manager.
- Create separate projects for each researcher and have them import (merge) their data into the master project at regular intervals.
- Ensure that each team member uses their unique user name and initials when accessing their project. Use this information to track the work of each researcher in the merged project.

In smaller projects, you may choose to share a single **NVivo** project file. In this scenario, it is still important that each researcher access the project using their unique user profile.

Only one user can access an **NVivo** project file at a time.

Creating and Cataloguing Themes and Topics

Where coding consistency is important, you will need to negotiate and agree on a node structure early in the project. Have regular discussions about node definitions and how the node system is evolving. In **NVivo** you can:

- Create a preliminary node structure in the 'master' project and have team members import it into their own projects.
- See which nodes have been created or modified and by which team member—do this in **Node List View** or by running a **Node Summary** report
- Use the **Description** field in **Node Properties** to define the use of the node so that all researchers have a common understanding.

While a common node structure is important for efficiency and reliability— it should remain flexible so that new insights and exciting ideas are not lost.

One method of retaining this flexibility is for each researcher to add a node to their project which is named just for them, e.g. 'Fiona'. Using this node as the parent (top level node), any new nodes they create can be stored under this parent node and coded to. When projects are merged together, even if two researchers have created child nodes with the same name, these nodes will stay separate and easily identifiable (because their hierarchical names are unique) so coding can be discussed and if required, the node(s) moved into the main node structure.

Coding the Data

If multiple researchers are coding the same data set, you may be interested in the consistency of their coding. **NVivo** provides a number of ways to check consistency or coder reliability:

- Run a **Coding Comparison Query** to determine the percentage of agreement and disagreement between coders.
- Display coding stripes for users— you can open a data source and see the coding done by each researcher.
- Filter the content of a node to see only the references coded by selected researchers

Remember that inconsistency in coding is not necessarily negative— it may prompt productive debate and deeper insights into the data.

Gathering Your Sources

Making Useful Qualitative Sources

Qualitative data is often rich, complex, multifaceted and vivid. Sometimes the challenge is how to reduce sources of data sufficiently to make sense of events. Sometimes the task is to manage the growing body of data, as you explore complexity in detail, further research events, create records and your understanding of them changes.

Your records will be more useful if you are aware of the tools **NVivo** offers for managing, describing and coding your sources of data from the start of the project. The sources of data in your project are represented within **NVivo** as 'Sources'.

The choice of sources to include in your project is very much determined by your research question or topic. In particular, while building up your sources, ask:

- How can you increase the richness and accessibility of your sources? (i.e. Should you include additional detail about the interview other than its direct transcript?)
- Can you work directly with your audio and video sources or should they be transcribed—if so, does the whole source need to be transcribed or are pertinent quotes enough?
- What contexts should be recorded? (i.e. Would your analysis benefit from additional detail about the source itself which you could record in descriptions and memos?)
- Have you included any demographic details or characteristics related to your data which may be of interest to your research? (i.e. Do you want to be able to distinguish between data related to different sites or groups of participants?)
- What tasks are mechanical and clerical, and can you use **NVivo** to do these automatically? (i.e. Can you format your sources to take advantage of auto coding?)

Working in Your Sources

The sources of data for your project may be gathered in a variety of different ways, be in a number of different formats or represent very different types of data. **NVivo** provides different types of sources which reflect these differences in your project's data:

- **Internals** typically form the majority of your source material. These may be your field notes, interview transcripts, audio or video material and so on.
- **Externals** can represent those sources of data which cannot be directly imported into your **NVivo** project
- **Memos** to record your own ideas, insights and growing understanding of your data

Naming and Describing Your Sources

Names matter! You will want to recognize a source easily. The longer the name, the harder it is to recognize and it is particularly difficult to handle many names that start with the same words (e.g. “interview with”). Pay careful attention to the naming scheme that you apply to your sources, and consider how to make use of folders to manage the different sorts of sources you have.

In the **Volunteering** Sample Project

> The documents or audio/video sources which contain interviews are stored in a folder called **Interviews**. The documents then include the name of the person interviewed.

You can use a source's description to record details that will assist your recall, offer important context or record reflective comments.

In the **Volunteering** Sample Project

The description of each source in the **Interviews** folder lists the name of the interviewer, the date and time of the interview and its social location.

Formatting Your Sources

It is important to carefully format and place content while preparing your data records, as this will significantly increase their readability and will also impact your ability to make use of functionality related to this structure. **NVivo** provides a number of ways to structure the content of your sources. These include:

- Paragraph Styles

  Different paragraph styles can be used to identify sections of content, especially if your sources are in any way structured.

  In the **Volunteering** Sample Project:

  - The focus group transcripts are structured by participant name (formatted as *Heading 2* style) and their comments (formatted as *Normal* style).
  - The interview transcripts are structured with the prepared questions formatted as *Heading 1* and the answers formatted as *Normal*.

  Using paragraph styles in structured documents enables you to:

  - Auto code your sources - make a node for each *Heading 1* and code the *Normal* text at the node. Refer to Auto Code by Paragraph Style for more information.
  - When exploring the content coded at a node, spread coding or view the context based on paragraph style.

  Making text large and bold does not mean it is a *Heading;* you must apply a *Heading* paragraph style.

- Paragraphs

  Careful paragraph structure within your sources' content allows:

  - The ability to spread your view or coding to surrounding paragraphs within a node or in query results.
  - The ability to 'auto code' your sources based on their paragraph structure.

- Fonts and Color

  You can add additional meaning to your sources’ content using fonts and color for the following purposes:

  - To visually identify different themes
  - To visually identify moderator, rather than participant comments
  - To visually indicate emphasis or importance
  - In a project researching the fine nuances of discourse, color and font could indicate hesitations, moods, double meanings etc.
  - To avoid creating nodes unnecessarily in the early stages of your project. Return and code the colored text once the categories are firm.
  - To identify different team members’ inserted commentaries

  You may wish to keep a key of the font and color combinations you are using for different purposes.

Making Links in Source Content

When you are reading or reflecting on content of your sources, there will be many occasions when you wish to record that you should "See Also" another source, or some of its content, or an item outside your project. By placing these links you ensure that a comparison will be remembered, or a relevant piece of evidence followed.

In the **Volunteering** Sample Project

'See Also' links are used in the following ways:

- The **Defining 'volunteer'** memo contains a 'See Also' link to the **Wesley Mission - Definitions of 'Volunteering'** external.
- The interview transcript document **Frederic** contains a 'See Also' link from different points of the interview when Frederic makes two very different comments about the nature of volunteering.
- Two of the participants, Anna and Grace volunteered in similar capacities. A 'See Also' link has been placed from the relevant quotation in the **Grace**'s Interview transcript document to the relevant quotation in **Anna**'s interview transcript.

Annotating Your Sources

An annotation can be placed in the content of any source to record brief comments attached to particular places in the content. Annotations are valuable for ensuring that your ideas about your data are stored at their context without interrupting the flow of your source. Their uses will differ across projects.

In the **Volunteering** Sample Project

Annotations are used in the following ways:

- The transcript document **Fredric** contains annotations with comments about this participant's choice of words in one instance and seeming contradictions in his opinions.
- The **Project Journal** document contains an annotation with observations about the comments of one researcher by another.

Working with Documents in NVivo

Depending on the nature of the sources of data you are working with, you may choose to either import your documents into your project, or create them within **NVivo**.

There are number of features which you can employ for different purposes while working with your documents:

- Annotate document content to record your reflections on particular selections
- Create 'See Also' links from one document's content to another - to indicate its relevance
- Create nodes and code as thoughts occur whilst reading through the document's content (This is particularly helpful when using field notes; as you type, you are thinking about the important categories you should create)
- Create nodes to gather the answers to each of your interview questions and review to determine whether the questions need rewording (This can be especially useful at the early stages of data gathering)
  In the **Volunteering** Sample Project

  The documents in the **Interviews** folder were auto coded by heading level to create the tree node **Topics - Interviews** and its children, **Q.1. Current use of time**, **Q.2. Time use ten years on**, **Q.3. Volunteer work means?** etc and gather the answers to these interview questions from all participants.

- Create a memo for each source to store a hunch or a fleeting observation, or the recalled context that was not in the interview transcript. The source and its memo can be linked together using a memo link.
  In the **Volunteering** Sample Project

  > The memo **Mary's Interview - Context** contains additional detail which may be relevant in the analysis of this source. This memo is linked to **Mary's** interview transcript document.

Handling Audio and Video Sources

The proliferation of digital video cameras and audio recorders has provided researchers with a rich and readily-available source of data. Like all other data sources, researchers need to consider how this new media will be gathered, organized and analyzed.

Gathering Video Data

If you are recording in a research setting— for example, an emergency room at a hospital or a school playground, you may want to consider the following:

- Do you require consent from all participants participating in the audio/video?
- Recording equipment can be distracting— take note of how the presence of the camera or recorder affects participants.
- Consider how much data you want to collect—hours of footage can be a daunting and unmanageable prospect.
- Keep a log of what you are doing— date and time, location, the events that took place, the surrounding conditions and your personal feelings about the events. In **NVivo**, this material can be imported as a transcript or as annotations and memos.
- Do you want to edit the audio/video material? You will need to do this before importing it into **NVivo**. Determine what editing software you will use and check the equipment manuals for information about downloading media files from a recorder to your computer.

Many ethnographic and social research projects ask participants to record their own experiences. If you are gathering this type of data, be aware that people have different levels of expertise in using video/audio equipment and you may end up with large amounts of unusable material. Ask participants to keep a log of what they are doing and why they chose to record particular events.

Organizing Audio/Video Data in NVivo

Like other sources in **NVivo**, you can use folders and sets to organize your audio/video data. Audio/video sources can be kept in any **Internals** folder— you can create a folder for all your audio or video sources or you can store them with related documents or pictures.

In the **Volunteering** Sample Project

> The source **Video - Non-volunteers** is stored in the **Focus Groups** folder with other documents. You can distinguish video sources from document sources by their icons.

If you are working with large or many video files, consider the options for storing the media outside of your **NVivo** project. Refer to Storing Audio and Video for more information.

Transcribing Audio/Video Data in NVivo

Do you want to transcribe your audio and video sources? Consider the following:

- Since you can code directly on an audio or video, you may not need a transcript.
- Without a transcript, an audio or video source is excluded from **Text Search** or **Word Frequency** queries (although you can query annotations instead).
- You can transcribe the media in **NVivo** or you can import transcripts from Microsoft Word.

- If you are importing transcripts from Microsoft Word, ensure that they are formatted correctly. Refer to Importing Transcripts for more information about valid formats.
- You can choose to transcribe only the 'interesting' portions of an audio or video.
- A transcript can be a record of what is said or descriptions of what is occurring. It may also be a simple collection of keywords.

Handling Picture Sources

In doing your research, you may find that some concepts are best explained using pictures. You may be using pictures to explain a concept to respondents or asking respondents to use them to illustrate their point. Or, you may be taking digital photographs to provide evidence of an event you are investigating. With the ease of point-and-shoot digital cameras and a whole digital library of images available for public use, researchers are enriching their research data through the inclusion of pictures.

Gathering Pictures

NVivo enables you to import images in various formats (.bmp, .gif, .jpg, .jpeg, .tif, .tiff) to create picture sources. Once imported, all images are converted into JPEG files and are also exported in this format.

When planning the pictures to include in your project, you may find it useful to think about the following:

- If you are including images taken from the Internet, check the copyright. Most digital images have watermarks that indicate if the image is copyrighted. These images are usually available for sale.
- If you are taking the photos yourself and publishing these as part of your research, permission may need to be obtained from your subjects, from the owners of a place, or from the organizers of the event you are researching. You may also need to get approval from your ethics board or its equivalent.

Checking these ahead of time will save you from wasting precious effort and will ensure that the pictures can be included in your research.

Organizing Your Pictures

Like all other sources, you can store pictures in **Internal** folders within **NVivo**. A good way to organize pictures is by using sets to create a gallery. You can create a theme for each picture gallery. Refer to Making a Picture Gallery for details of how you can do this in **NVivo**.

Analyzing the Whole or Part of a Picture

When analyzing a picture, you may find that there are certain portions of the image that are of particular interest or value to your research. **NVivo** enables you to select regions of a picture and include notes or log entries.

Like all other source types, you will be able to code picture sources and query them during your analysis. Both the picture and the log entries can be annotated, and linked to other documents within the project. In the log entry, you can also add a website link.

Handling External Sources of Data

Many projects make use of sources of data that cannot be imported into your project. These sources of data can be represented in **NVivo** using externals. Externals are simply sources representing data that will remain outside your **NVivo** project.

Externals could be used to represent:

- Web pages which can be accessed using your internet browser
- Electronic files stored on your computer or a computer network - such as PowerPoint presentations
- Data records which are far too large to include in a project, such as historical records, books or large reports

Structuring Externals

An external has both its properties (i.e. type, file path, location description) as well as its contents. These contents can be typed into the external and structured using paragraph styles, paragraphs, fonts and color in the same way as documents and memos.

Additionally, when creating an external you can nominate a unit, start and end range to create a named paragraph for each unit. You can then enter your comments or transcription at the relevant points. This external content can be reviewed and separately explored, and like any other source can be edited, annotated, linked, coded and searched.

In the **Volunteering** Sample Project

Externals are used in the following ways:

- The external **Volunteers - How to get them, How to keep them** represents a book. The units in this external's properties are chapters and the start and end range is the number of chapters relevant to this project. The contents of the external contains a chapter summary for each of these chapters.
- The externals **Wesley Mission - Definitions of 'Volunteering'** and **Cross Cultural Solutions Website** represent web pages. The external content is summary information from these web pages identified as pertinent by the researchers. To open the web pages, on the **Project** menu click **Open External File**.

Using Externals in Conjunction with 'See Also' Links

By creating an external to represent and link to an electronic file or web page, you can then add any number of 'See Also' links to this external from within the content of your sources.

Most importantly, if the URL is changed or you move the electronic file that is represented by this external, you need only update the link to this file's external, rather updating each hyperlink to it within your project.

It is highly recommended that this technique be used for files that are frequently linked to within your project.

Capturing Ideas with Memos

Memos provide a way to store and efficiently retrieve ideas, insights, interpretations, growing understanding or recognition of puzzles that have to be understood within your data. You may also wish to write analytic, methodological or theoretical memos and keep them separate from, but linked to, the sources of data from research events such as interviews or observations. Those memos grow, are themselves data and can become important sources within your project.

Evolving Memos

Memos can 'grow' into documents, or documents into memos. You can redefine the sources in your project as you make decisions about their role. You might for example write a document recording field notes of an observation. It can grow into a more reflective piece about the attitudes of participants and their significance in your study, so you can change it into a memo.

Linking Memos to a Source or a Node

You can choose to link a particular source or node to its 'own' memo, indicating that the ideas, insights, reflections and information in its content are linked to that item. At any stage when you are considering the item, you can find and add to or edit its memo to store your reflections or discoveries.

In the **Volunteering** Sample Project

Memos have been linked to other items for the following reasons:

- The memo **Defining 'Volunteer'** contains the researchers' reflections about the different meanings of this term. Given its relevance, this memo is linked to the tree node **Meanings of Volunteer Work** using a memo link.
- The memo **Mary's Interview - Context** contains additional detail which may be relevant in the analysis of this source. This memo is linked to **Mary**'s interview transcript document.

Making Nodes

What are Nodes?

Your project needs places not just for its data sources but for the ideas that you bring to it and themes you generate from the data. As you work with your sources, you will gather and explore existing and new categories for thinking about them.

Nodes are containers for these ideas within your **NVivo** project and contain the evidence within your sources supporting them. Creating and exploring nodes is a way to think 'up' from the data and arrive at higher level explanations and accounts.

Coding is the process by which you nominate a portion of a source which relates to a node.

Different Types of Nodes

NVivo provides different types of nodes which are suited to different types of ideas and concepts you are likely to represent in your project:

- **Free Nodes** can be used as containers for 'loose' ideas which are not conceptually related to other nodes in your project. As your project develops, these may be moved into a logical place in your tree nodes.
- **Tree Nodes** can be used to represent the concepts and categories in your project which are logically related as they can be organized in a hierarchical structure (i.e. category, subcategory)
- **Cases** represent the entities within your research (i.e. people, schools, institutions, families). These can also have 'attributes' to record the characteristics of those entities that you want to ask about. Cases, like tree nodes can also be organized hierarchically.
- **Matrices** can be used to show how the contents of different nodes relate to each other. They are created by querying your data using matrix coding queries and are presented in a tabular format.
- **Relationships** represent what you know or discover about relations between items in your project.

You may have determined some of the themes or topics you want to represent as nodes in your project prior to exploring your sources, perhaps as a result of your prior investigation into the literature or the key areas in your interview questions. As you work through your sources, you may find a number of additional themes or topics that you also want to represent as nodes.

Strategies for Creating Nodes

If you are unsure as to which type of node to create, ask these questions of what you aim to represent in your project:

- Does it represent an entity within your research, such as a person or institution?
  Consider creating a ***case*** node
- Does it represent a topic, theme or category within your sources? If so, does it relate to other topics, themes or categories within your project?
  If it does, consider creating a ***tree*** node. If not, consider creating a ***free*** node.
- Does it represent a connection between items in your project?
  Consider creating a ***relationship*** between these items

Naming and Describing Nodes

Names of nodes, as of any project item, are best kept short and very pertinent. This will help you find the relevant node quickly for coding and will help to ensure that you use it consistently.

A description is highly useful for recording current thinking on a concept, or instructions about how the node is being used.

In the **Volunteering** Sample Project

> Each of the parent tree nodes (e.g. A**ssumptions**, **Contexts**) contains a description about what they are intended to represent and the nature of the data to be coded at them.

Nicknames can be provided to provide rapid access to frequently used nodes while coding.

In the **Volunteering** Sample Project

> The tree nodes with the most coding **Health and Welfare** (within the **Contexts** tree node), **Giving Your Time** and **Unpaid** (within the **Meanings of Volunteer Work** tree node) have nicknames so they could quickly selected whilst coding.

How Many Nodes Do I Need?

It is important to find a balance between the nodes in your project being too sparse to do justice to the data and too many as to be unwieldy and onerous to use consistently. These factors may influence the number of nodes in your project:

- The nature of your research:
  A short term project that aims for a thematic summary of the material, might have relatively few nodes compared with a long term project whose goals include detailed interpretation, pursuit of multiple meanings, or pattern seeking and validation across samples or sites.
- The nature of the particular research methodology for your project (i.e. a discourse analysis project may use only annotations and memos to store interpretations of sources, rather than nodes.)
- The stage of your project:
  The number of nodes will often change throughout the course of your project. You may start with very few categories and themes in your initial topic coding, then create a number of new categories and themes as you continue with your analysis and then refine them down in the later stages.

Building Efficient Node Hierarchies

Organized hierarchically, nodes show the shape of the emerging categories, the concepts in which the data are being analyzed.

A well organized hierarchy of tree nodes or cases should:

- Allow you to locate a particular node rapidly
- Help in recognizing a particular node's relation to other nodes and its place in your project
- Provide the ability to see and seek the dimensions of the category represented by a node (Sometimes referred to as "dimensionalizing").

The simple rule is if a node obviously "fits" in your conceptual tree, place it there. If it doesn't, don't force it—create it as a free node until its relation to other nodes becomes clearer.

Structuring Your Node Hierarchies

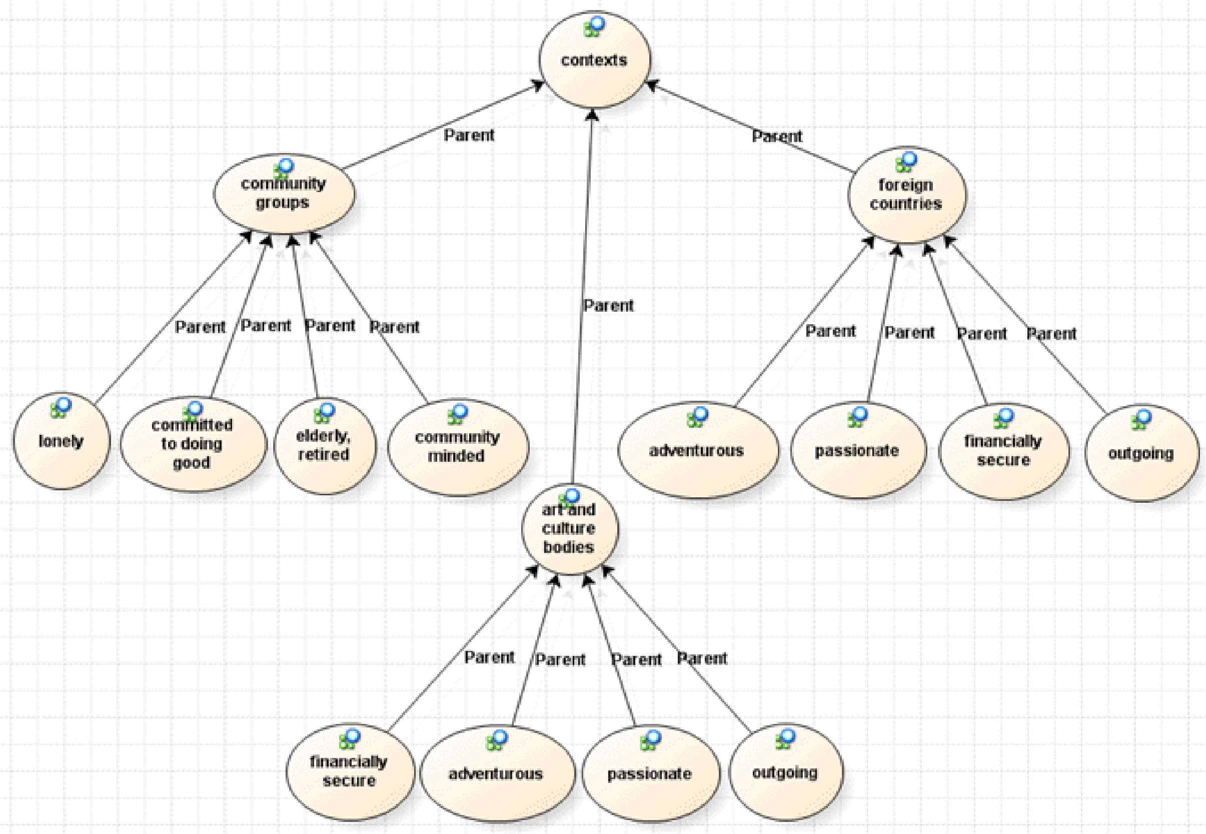
Node hierarchies are usually best kept coherent and concise. Ensure that conceptually related categories are kept together and unrelated categories apart.

The most common problem hierarchy is one in which a node has been made for every possible combination of categories, rather than coding the same passage at multiple nodes.

If you are unsure whether a category fits into an existing node hierarchy, ask whether it is a sort of the category in question. If it is, it should create a child node within that tree, if it is not, consider whether it should be part of another tree or exist independently (i.e. as a free node). Similarly, if you notice that a node name appears more than twice within your trees, you may need to create it as its own node.

In the **Volunteering** Sample Project

> A number of contexts in which people volunteer (community groups, foreign countries, art and culture bodies, etc.) are discussed in interviews and focus groups.
>
> Also discussed were the varying images that these participants had of volunteers (adventurous, committed to doing good, lonely, passionate etc.) in these contexts. For instance, were volunteers in community groups perceived differently from those volunteering in foreign countries?
>
> These categories could be represented using tree nodes as follows:

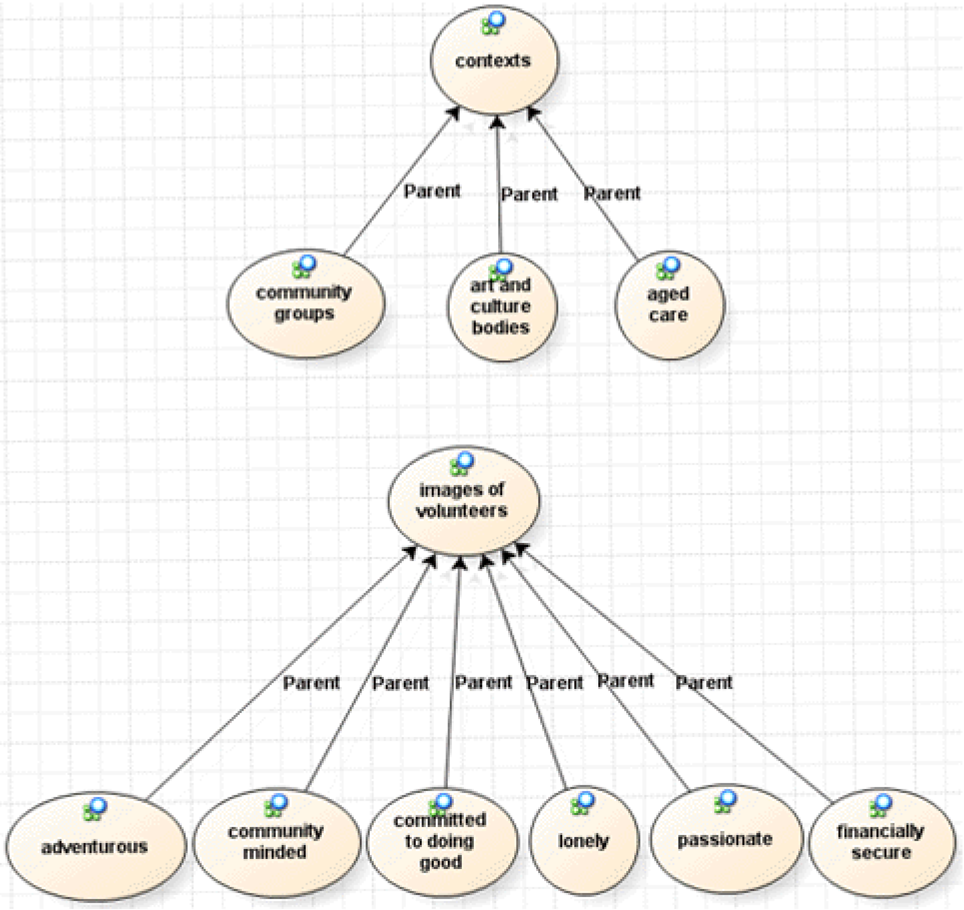


This node hierarchy promises four very serious problems:

- It will grow rapidly as more images of volunteers are identified within the sources (all images must appear under each node for a context) or more contexts studied (the new context must appear under each image).
- It will impede category development (i.e. it would be difficult to add further dimensions of images, or different types of context).
- It will make it difficult to ask more general questions (such as in which contexts are volunteers considered lonely or elderly/retired).

- It will also confuse your thinking, since contexts of volunteering are different sorts of things to images of volunteers.

The optimal approach would be to create a tree for volunteering contexts and another tree for images of volunteers and to code text at both the relevant context and the relevant image.



You can then use queries to find content which is coded by particular combinations of nodes.

Using Cases and their Attributes

Cases are nodes which represent the entities within your research (i.e. people, schools, institutions). Your cases should gather together all of the content in your sources related to specific entities (i.e. individual research participants, schools, businesses). For instance, a case for the participant **Bernadette** should contain references to the transcript of her interview, any contributions she made in the transcript of a focus group, as well as any comments that other participants made about her.

In the **Volunteering** Sample Project

> Cases represent each of the interview and focus group participants and contain references to the content of the sources related to each individual.
>
> For interview participants, each case (e.g. **Mary**) contains a single reference to their interview transcript. The cases for the focus group participants (e.g. **Annette**), on the other hand, contain a number of references. These references are to each of the comments that participant made in that focus group.

What is an Attribute?

Cases can have 'attributes' to record those characteristics of the entities which may prove relevant when analyzing the categories and concepts emerging from your data. If you are interviewing, you may have information about many attributes of a respondent (i.e. age, gender, education level) that will be highly relevant.

Each attribute in your project can have many values. For example, the attribute **Age Range** may have the values of **20-29**, **30-39**, **40-40** etc. These represent the range of values for that particular characteristic within your project. To store information about your cases, you assign the relevant value for each attribute to each case.

In the **Volunteering** Sample Project

> The attributes are all those characteristics of the participants which may influence their opinions and views about volunteering. These include their **Age Group**, whether they have **Current paid work**, and whether they have **Ever done volunteer work**.

Why Record the Attributes of Cases in my Project?

Cases and attributes are ideal for asking comparative questions about the nodes representing other concepts and categories in your project. They enable you to compare and contrast the contents of your cases based on the attribute values assigned to them (i.e. Do interviewees under the age of 30 have different perceptions of volunteering to those over 50?). Such comparisons can lead you to new understandings of your data and the categories and concepts emerging from it.

When working with coding stripes in a source or a node, you can choose to display the stripe for a selected attribute. For example, you could display a stripe for *female* and see all content coded at cases that are female.

Representing Relationships between Items

Relationships record statements or hunches you have developed about how items in your project are linked. You may represent as relationships the ways in which your research participants are related (i.e. **Anna** lives with **Sunil**), how concepts are related (i.e. **Context of volunteering** impacts **Images of volunteers**) or any other links between items in your project.

In the **Volunteering** Sample Project

> It became apparent that the context in which they volunteer impacted the images of volunteers reported by the participants.
>
> These contexts and images of volunteers had previously been recorded in the project as tree nodes, so a relationship was created between them with a type of **impacts**. Any evidence for this relationship was coded at it.

Classifying Relationships

To record relationships between items in your project, you need to indicate the nature of those relationships by creating relationship types. These relationship types classify the relationships within your project and allow you to make comparisons between all the relationships of a particular type to another. Relationship types have both a name and a direction.

Naming the Relationship Type

It is a good idea to name relationship types with a verb or verbal expression, like "is friends with". Verbal relationship type names assist because they make it clear what is being stated, clarify the relationship type's direction and what items the relationship is from and to.

In the **Volunteering** Sample Project

The relationship **Bernadette is friends with Ken** is a verbal expression. This makes it easy to determine who is friends with whom. If it was expressed using a noun (ie. Bernadette friend Ken), the nature of this relationship would be difficult to determine.

Relationship Type Directions

When creating a relationship type, you must give it a direction. There are three types of relationship type direction, each giving different meanings to the relationships you create:

1. **Associative (or no-direction)**: An associative relationship can be used to demonstrate that items are in some way affiliated. Associative relationships between two items must always hold both ways (i.e. Anna lives with Sunil, Sunil must also live with Anna).
2. **One-way**: A one-way relationship can be used to demonstrate a relationship between items which has a definite direction, an agent and a recipient. A one-way relationship but doesn't have to hold both ways, it may or may not. (i.e. Bernadette likes Ken, Ken may also like Bernadette but may not and that doesn't change the fact that Bernadette likes him.)

   In the **Volunteering** Sample Project

   A relationship exists between the contexts in which people volunteer and how people see those volunteers (i.e. their image).

   This is represented using a one-way relationship illustrating that the context of volunteering **impacts** on the images of those volunteering, but the reverse is not necessarily true. How people see volunteers does not seem to impact on the context of the volunteering.
3. Symmetrical (two-way): A symmetrical relationship demonstrates some sort of two-way activity between the items. These relationships imply they hold both ways, such as "being married to" or "being a sibling of".

Evolving Nodes

Nodes express your growing understanding of the concepts and ideas within your data. Of course, your understanding of the nodes in your project may change over the course of your analysis. You can represent this change within your project using any of the following functionality:

- Changing a node's type
  This can be especially useful for free nodes that were created when they had no logical place in a hierarchy of trees until later in the project.
- Changing the node's position in a hierarchy
  A node's position in a hierarchy is a significant indicator of its relations to other categories. As you explore data and create categories, you may wish to adjust its position in its hierarchy or move it into another hierarchy.
- Merging nodes
  If you find nodes which have the same meaning and you wish to simplify the system by combining them, they can be merged.
- Deleting nodes
  You may have created a node early in your project for a category or theme which you thought might be important, only to find out as you work through your sources that it wasn't. This node could be deleted to simplify your nodes.
- Changing a node's name
  As you work through your data, you may want to change the names of your nodes to reflect changing understanding of its category or theme. If you do this, consider whether you also need to revise its coding.

Coding

What is Coding?

Coding is the process of bringing together passages in your data that seem to exemplify an idea or concept (represented in your project as nodes). As such, coding is a way of abstracting from your source data to build a greater understanding of the forces at play.

In **NVivo**, coding involves identifying references to different ideas, concepts or categories within your sources and linking them to the nodes which represent them.

> Remember that you can code any portion of a sources' content to any number of nodes to demonstrate that it relates to each of their concepts or categories.

Why Do I Need to Code My Sources?

Coding the content of your sources can contribute to your analysis in the following important ways:

- Coding generates ideas
  While coding your source material, you may interpret passages and see new meanings in the data.
- Coding allows you to gather together and see all the material related to a category or case throughout all of your sources.
  Seeing all this material allows you to review coded passages in context and create new, finer categories as you gain new understanding of the meanings in the data.
- Coding your sources facilitates the seeking of patterns and theories.
  The coded content of your sources can be searched using queries and find functionality to test theories and find new patterns within your data.

Approaches to Coding

In different stages in your project, the way you approach coding the content of your sources may change. These approaches have different purposes and contribute differently to the process of analyzing your data:

- **Descriptive coding** is the process of identifying information that describes the cases in your project. This process relates both to the coding of information at cases and the creation of attributes to classify them.

  NVivo provides a number of ways to expedite descriptive coding

  In addition to coding by selecting source content and assigning it to a new or existing case, you can also:

  - Create cases whilst importing sources
  - Code an entire document at a case
  - Create a case from a document
  - Import the casebook from a spreadsheet

- Auto code by heading level when the headings in your sources reliably indicate the content to be coded and cases to be created or coded to In the **Volunteering** Sample Project

  The documents in the **Focus Group** folder are structured with each group participants' name in a heading paragraph style and their comments in normal paragraph style beneath them.

  The document **Non Volunteers** was auto coded by heading level to create the cases, **Elaine**, **Raul**, **Stephanie** and **Roberta**.

  Each of these cases contains all of the responses that these participants gave in this focus group as outlined in the document.

- **Topic coding** is the process of assigning references within your data to the topics, categories or concepts they relate to. It is necessary to see all the information about these topics, categories or concepts in your data to facilitate further analysis.

NVivo provides a number of ways to expedite topic coding

In addition to coding by selecting source content and assigning it to a new or existing node, you can also:

- Auto code all sources formatted with a specific heading or paragraph structure:
- Auto code by heading level when the headings in your sources reliably indicate the content to be coded.

In the **Volunteering** Sample Project

The documents in the **Interviews** folder are structured with each question in a heading paragraph style and the answers in normal paragraph style beneath them.

These documents were auto coded by heading level to create the tree node **Topics** - **Interviews** and its children**,Q.1. Current use of time**, **Q.2. Time use ten years on**, **Q.3. Volunteer work means~**, etc .

Note: The question mark at the end 'Q.3. Volunteer work means?' has been replaced with a tilde (~), because node names cannot contain question marks.

Each of these child nodes contains all of the answers in each of the documents in the "Interviews" folder given under these topic headings.

- Auto code by paragraph when the selected sources have exactly the same paragraph structure. For example, the content in the second paragraph of each of the sources is the answer to the same survey question.
- Select whole paragraphs to code at particular nodes using code paragraph ranges
- Explore your sources content using text search queries and save results as a node
- **Analytical coding** is the process of interpreting and reflecting on the meaning of the data to arrive at new ideas and categories. This process entails gathering material that should be rethought and reviewed given your growing understanding of the categories in your data.

NVivo provides a number of ways to change the way you look at your data and assist this analytical coding

In addition to coding by selecting source content and assigning it to a new or existing node, you can also:

- Open a node and code its content at other nodes or create new ones (also known as "coding on")
- Open a source and display coding at particular groups of nodes, using coding stripes or highlight coding, to determine if you should code it further
- Create new nodes using In Vivo coding

- Explore your nodes content using coding queries and save results as a node
- Explore your nodes content using matrix coding queries and save results as a node
- Explore your nodes content using compound queries and save results as a node

Working with Coded Data

Coding brings together the material you have interpreted as relevant to a topic or category. Now, you can think about the category as you read and rethink that material. Are there two different sorts of material coded here? Can you see ways to develop your understanding of the passages coded here?

The first coding is often a first step to much more productive reflection and analysis. As you reflect, you may want to refine your coding, and as you discover more meanings in the coded data, you may want to move on to create and code at more subtle categories and subcategories.

Reviewing a Node's Content

When you open a node, you can see all passages within your sources which have been coded at it. This allows you to:

- Read and review everything coded at this node
- View coding stripes to show the coding at other nodes for any passage
- Return to the source of any of the coding references to explore that context, and perhaps do more coding or annotating of that source

Altering a Node's Coding

Coding is rarely a one-stage process. As you review the coded data at a node, you will often see ways to improve your coding. You can code a node's content in the same ways as coding a source's content. As you review a node's content, you may want to:

- Expand a particular coding reference to include relevant context so this will be more easily recognized later
- Remove some of its coded content by 'uncoding' it
- Develop ideas by coding content to other nodes. This process is termed 'coding-on' as it takes the coded data onwards to expand existing nodes or code at new, refined categories.
- Create or shift nodes and rethink them as you respond to what you are seeing in the recontextualized, coded text. Data gathered at a node offers a new understanding of the category and this often leads to development of new dimensions.

Comparing Coding

It is critically important that your nodes are being coded to reliably. If you work in teams, it is important to ensure that all team members have the same understanding of the topics or categories that your nodes represent and importantly that this understanding does not change over time.

Usually the requirement is not that team members code identically (as coding reflects their different interpretations) but that they reliably identify and deal with inconsistencies in the interpretation of nodes and the style of coding.

Comparing coding in NVivo

> To compare coding, import two copies of the same document into your project to be independently coded by two team members. Run the **Coding Comparison Query** and assess the percentage agreement and disagreement.
>
> Pay attention to the differences in the nodes used, the passages coded and the spread of coding selected.

How Much Coding Should I Do?

It is very easy to do too much coding. Coding can become a problem because it is easy to do, when more analytical processes may seem difficult. You can code particular content within your sources as many times as you wish, and at as many nodes as you create. So you need to monitor and judge the usefulness of your coding.

Coding must have a purpose—it is never an end in itself. Watch for the following indicators that you may need to re-evaluate your coding strategy:

- The purpose of the coding is hard to discern
- The coding is largely descriptive (this text is about that topic) and not often more analytical (this issue obviously matters—why?)
- You are easily bored whilst coding. If coding is generating new insights and questions, you won’t be bored
- You are doing nothing else. Coding is only one way of expressing interpretation, and if you are engaged in purposive coding, it will be interspersed with other tasks like writing memos and querying the data

If you need to control the amount of time spent coding, but you are not confident that you can safely cut corners, a useful strategy is to do very broad-brush coding at general categories. Should these prove important later, you can return to work with the coded data in each node and code on to make more subtle nodes as required.

Querying Your Data

Asking Questions of Your Project

In the process of your analysis, you may want to search, filter, focus and selectively interrogate your data, rather than always working with all of your data. Asking questions from the very beginning of your project before you have recorded all your data or coded sources fully can usefully inform redesign of your project (Do I need to gather additional sources of data? Should I re-frame the questions in that section?).

Different Types of Questions

You will need to ask a variety of different questions of your data throughout your project. Different types of questions are best addressed using particular types of **NVivo** functionality, for example:

- If you would like to find *items* based on their name, particular characteristics or associations with other items in your project, consider using ***Find***.
- If you would like to find *content* with particular text or coding characteristics, consider creating a ***Query***.

How to Set Up My Questions in NVivo?

It is important to have a very clear idea of what you are asking to know which **NVivo** functionality to use and how to use it. The following steps may assist you:

1. State the common-language question you wish to ask
2. Select the appropriate functionality for that question
3. Restate your question in terms of the processes it requires. Step logically through each process
4. Carefully state the answer, accurately recording what it tells you

In the **Volunteering** Sample Project

Adventure was discussed by the interview participants, but was adventure discussed in any of the memos? Let's re-frame this question:

Common language Question: Do any of the memos discuss volunteering in terms of adventure?

Appropriate Functionality: The tool is Find (I want to find items rather than specific content)

Restated Question: Find memos where any data is coded at the node for adventure

Answer: A list of all the memos that were coded at the **Adventurous** node (in the **Images of Volunteers** tree)

Strategies for Finding Items in Your Project

You may want to find particular items in your project for a number of reasons. You may want to work with those items by adding them to a model or set, or use them as the basis for further analysis. For example, you may want to add the items in a relationship with the "Images of Volunteering" node to a model or a set. On the other hand, you may want to find out how many men have never volunteered.

Different types of find functionality are available in **NVivo** to address these different types of questions about your data.

- If you would like to find items based on their name, consider using ***Find***.
- If you would like to find items with particular characteristics, consider using ***Advanced Find***.
- If you would like to find items which are associated in a particular way with other items in your project, consider using ***Grouped Find***.

Find - Finding by Name

Find allows you to locate items with a particular name or portion of a name. This can be helpful when you have a complex folder or hierarchical structure with a large number of items, or when you would like to find items of different types but the same name.

In the **Volunteering** Sample Project

> A document and a case both have the name **Mary**. You can locate both of these at once using Find.

Advanced Find - Finding by Feature

Advanced Find helps you locate items with certain characteristics. This may be helpful in asking preliminary questions of your data. You might ask whether you have any cases of employed men who volunteer to determine whether this is an area for further analysis or data gathering. Or you may ask what are all the items which have a see also link to this document?

In the **Volunteering** Sample Project

> In the **Volunteering** Sample Project, you might use **Advanced Find** to retrieve all of the relationships involving the **context** tree node to add to a model. You could use **Advanced Find** to retrieve all of the female cases in a particular age range.

Grouped Find - Finding by Association

Grouped Find helps you find items which are associated in a particular way with other items in your project. That is, you might want to see each of the nodes that code each focus group transcript document.

In the **Volunteering** Sample Project

> You may want to review how the different meanings of volunteer work occur in each of the focus groups.
>
> Using Grouped Find, you can list each of the nodes that code each focus group transcript by looking for items coding the **Focus Group** folder (scope) within the **Meanings of Volunteer Work** tree node (range).

Working with Queries

Throughout your project, you will need ways of discovering and exploring patterns, testing hunches and creating and validating theories. You may want to discover words in the text that indicate patterns or themes and to discover and test relationships between the categories you have been coding at.

The nature of your question determines which of the different query types to use:

- If you would like to see all occurrences of a particular word or phrase in your documents, consider creating a **Text Search Query**.
- If you would like to see source content coded by a specific combination of nodes, or combination of nodes and attributes, consider creating a **Coding Query**.
- If you would like to see patterns in the source content coded by one group of nodes by another group of nodes, or one group of nodes by a number of attribute values, consider creating a **Matrix Coding Query**.
- If you would like to see source content that has been coded by a specific nodes and also has specific text, consider creating a **Compound Query**.
- If you would like to compare the coding of two researchers or research teams, consider creating a **Coding Comparison Query**.

If you want to search only specific groups of items, you can achieve this by selecting these items as the query's ***scope***. This can help you ask more targeted questions such as "Where is this word used *in these documents*?" or "*Of the content in these sources*, what is coded at the free node **Motivation** and the node **Sense of Achievement** (in the **Personal Goal** tree).

Text Search Query

As you work with your data, you may find that particular words or phrases are being used in different ways. So, you might want to see all of the instances of these word or phrases to see how often and in what contexts they appear. A Text Search Query allows you to do this.

You could also use Text Search Queries to:

- Explore the different use of words or phrases and their meanings in different contexts
- Gather and code material early in your project. Later, when meanings become clearer, you can review and code on to finer categories.

> Be aware if you are using text search code that it may miss out some useful references (e.g. if the specific words searched for have not been used) and that it may gather references not needed (e.g. where the researcher rather than the respondent used the word). Nodes created by text search are not a substitute for nodes created and coded to by yourself.

In the **Volunteering** Sample Project

In the **Volunteering** Sample Project, one of the questions that came up in reading through the data was "What motivates people to volunteer?".

The Text Search Query **Motivation or Reason** was created to gather all of the instances of the words 'motivation' or 'reason' and the paragraph surrounding them and code them to the free node **motivation**.

Not all of the instances of these words are relevant to our question of "What motivates people to volunteer?" Some were used in the wording of interview or focus group questions, and as such were not relevant and uncoded from the **Motivations** free node.

Coding Query

Once you have created nodes to represent the themes and categories in your data and coded the content of your sources to those nodes, you are likely to want to see if there are patterns in your coding. You can achieve this using Coding Queries.

Coding Queries can also be used to:

- Gather material coded in combinations of categories to see what new meanings emerge, then rethink and recode them
- Clarify a concept by asking where coding at this concept overlaps with coding at another or what is coded at some concepts but not others
- Sort through your coding by groups defined by your attributes

In the **Volunteering** Sample Project

> One of the questions of interest is "Do older volunteers have a personal goal of social interaction?".
> This opinion was expressed by participants who saw an image of an older volunteer and assumed she volunteers for social interaction, but is this demonstrated in the responses from older participants themselves?
> The Coding Query **Greater than 50 yrs and Social Interaction** gathers all content coded by cases where the attribute **Age Range** has a value greater than **50-59 yrs** and coded by the node **Social Interaction** (in the **Personal Goal** tree).

Matrix Coding Query

Matrix Coding Queries are a way of asking a wide range of questions about patterns in the data and gaining access to the content that shows those patterns. Matrix Coding Queries allow you to 'break down' one grouping of project items by another grouping of project items.

You can also use Matrix Coding Queries to:

- Discover and explore patterns early in your project
- Establish the strength of a pattern later in your project and analyze in detail the content of each cell
- Pursue themes broken down by significant factors. Read first what the men say, then what the women say—can you discern noticeable differences?

In the **Volunteering** Sample Project

> One of the questions that emerged through the interviews was "Do the images of volunteers differ with the context of their volunteering?".
> Given that the relevant passages were coded at the nodes in the trees **Images of volunteering** and **Contexts** where appropriate, content coded at these nodes can be retrieved using a Matrix Coding Query.

The Matrix Coding Query **Images of Volunteers and Contexts** was created to see the different images of volunteers and the different contexts and the content coded by both.

Compound Query

Compound Queries combine the functionality of Text Search Queries and Coding Queries. They allow you to find source content that has been coded by a specific nodes and also has specific text. They also allow you to find particular text which has a particular proximity to other text.

Compound Queries allow you to:

- Explore hunches about the use of particular words or phrases in content which is coded at nodes representing particular categories or concepts
- Rethink and refine a category within your project based on the different use of words or phrases within them
- Gather material where particular words have been used close to other words to see if there is a pattern emerging around their use

In the **Volunteering** Sample Project

> Some of the images of volunteers depicted people volunteering in foreign countries. One of the questions that emerged was "Do the young people see volunteering in foreign countries as exciting?".
> Given that the statements were coded at the **Foreign countries** node (in the tree node **Contexts**).
>
> The Compound Query **Foreign countries and excite** gathers all content coded by cases where the attribute **Age Range** has a value equal to **20-29 yrs** and coded by the context tree node 'foreign countries' and containing words beginning with 'excit'.

Coding Comparison Query

A coding comparison query enables you to compare coding done by 2 users or 2 groups of users.

It provides two ways of measuring *inter-rater reliability* or the degree of agreement between the users: through the calculation of the percentage agreement and *Kappa coefficient.*

In the **Volunteering** Sample Project

> Two researchers - identified by their initials **ST** and **MWO -** have coded the sources in the **Compare Coding** set.
>
> A **Coding Comparison** query has been created to check the coding consistency of these two team members - it is called **Coding Consistency for ST and MWO**.
>
> The query compares all the coding that ST and MWO have done in the set of sources.

Coding queries only search the coded content in your project. If you have only sparsely coded your sources or used nodes inconsistently, your coding query may not return the results you expect. This may not mean there is no association between these concepts in your data, just in their coding.

Handling Query Results

You have a number of options when choosing what to do with your query's ***results***. The options you choose are very much driven by the reason you ran the query.

- If you wanted to gather together material containing specific text or combinations of coding at different nodes, consider ***saving the results as a node.***
- If you wanted to locate items based on a specific criteria so that you can further analyze them as a whole, consider ***saving the results as a set.***

You can then use the node or set you created from these results as the scope of another query. So you are building more subtle enquiry on the results of your first question.

Using Charts

What Data Can I Chart?

A chart is an effective way of presenting and exploring data. The key to maximizing its usefulness is understanding which data in your research can best be represented or explored using a chart.

NVivo gives you seven types of data you can chart for analysis:

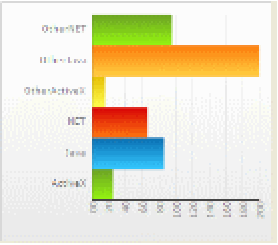
Chart Data Type	Description	Use this chart to find out:
Coding for a source	Displays coding information for a particular source.	• which nodes are most/least used to code this source • how much coding is done on this source • which users coded/did not code this source
Coding for a node	Displays how a particular node is used in coding sources. .	• which sources are most/least coded at this node • at this node, how much coding is done on these sources • which users coded/did not code these sources at this node
Cases by attribute value for an attribute	Displays the number of cases that match each value for a particular attribute	• which attribute value is matched to the most/least number of cases
Cases by attribute value for two attributes	Displays the number of cases that match various combinations of two attributes	• which combination of attribute values are matched to the most/least number of cases
Coding by attribute value for a node	Displays how a particular node is matched against one or two attributes and their corresponding combinations of attribute values	• for one selected attribute, which attribute value: - has the most/least percentage coverage - has the most/least number of coding references - is matched to the most/least number of cases • for two selected attributes, which combinations of attribute values: - have the most/least percentage coverage - have the most/least number of coding references - are matched to the most/least number of cases coded to this node

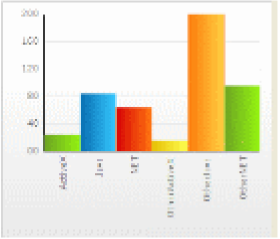
Coding by attribute value for a source	Displays how a particular source is coded against one or two attributes and their corresponding combinations of attribute values	• for one selected attribute, which attribute value: - has the most/least percentage coverage - has the most/least number of coding references - is matched to the most/least number of cases • for two selected attributes, which combinations of attribute values: - have the most/least percentage coverage - have the most/least number of coding references - are matched to the most/least number of cases coded in this source
Coding by attribute value for nodes	Displays multiple nodes matched against a particular attribute and its set of attribute values	• for a particular attribute and selected nodes, which attribute value - has the most/least percentage coverage - has the most/least number of coding references - is matched to the most/least number of cases
Coding by attribute value for sources	Displays multiple sources as coded against a particular attribute and its set of attribute values	• for a particular attribute and selected sources, which attribute value - has the most/least percentage coverage - has the most/least number of coding references - is matched to the most/least number of cases
Matrix	Displays the result of a matrix query in a chart	• the high/low points in the matrix data • any patterns or trends that may be present

What Type of Chart Should I Use?

There are numerous types of charts you can use to present research data. **NVivo** helps you make an effective choice by listing only the most relevant types to use for the data you want to chart. In general:

Bar or Column charts are useful when comparing quantity or analyzing an increasing/decreasing trend.





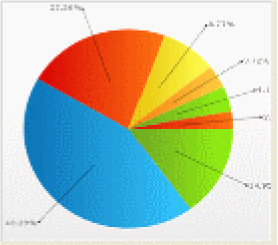
Bar

Column

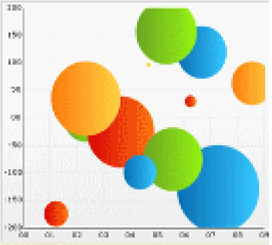
These types of charts can be further displayed as:

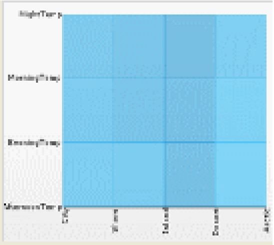
- 3-Axes bar or column chart
- Stacked bar or column chart
- Grouped bar or column chart

Pie charts effectively show the measure of different parts that make up a whole.



Bubble charts or **Heat Maps** show varying density of data when comparing combinations of variables or matrices.

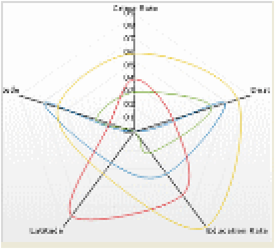




Bubble

Heat

Radar charts effectively display direction or trend when comparing several variables



Models

Modelling Your Project

Models are a way of illustrating and clarifying your ideas and of allowing you to reflect on different ways of seeing your data. During your analysis, models may be useful as aids to seeing links between concepts or items in your project, or as ways of reporting and demonstrating them.

Consider using models for the following purposes from the earliest stages of a project:

- To set out and review initial ideas, hunches or theories about the topic or question of your project

In the **Volunteering** Sample Project

The **First Thoughts** model was constructed prior to the coding of the data and depicts the factors influencing individuals' tendency to and engagement in volunteer work based on the researchers' understanding of the literature in that area.

- To visually represent the relationships between project items
- To identify emerging patterns and hunches, theories and explanations
- To provide a record of stages in a project

In the **Volunteering** Sample Project

The **First Thoughts** model was constructed prior to the coding of the data and illustrates the researchers' understanding of the possible factors at play based on their investigation of the literature. It contains only shapes and no project items as it was created before the project's nodes.

The **Coding Stage 2** model was constructed to following the topic coding of the focus group and interview transcripts. It illustrates the growing understanding of the factors identified in the data, given the understanding of the body of data. This model was converted to a static model as its intention is to show the understanding of the data at that particular point in the project.

The **Coding Stage 3** model was created following more analytical coding and in depth understanding of the data. It is a dynamic model (i.e. it contains links to project items) as its intention is to provide a different view of the information contained within the project.

- To 'zoom in' on a specific group of themes to explore other possible ideas, connections and relationships

In the **Volunteering** Sample Project

The **Perceptions** model provides a 'zoomed in' view of all of the different images of volunteers. Custom groups have been added to show whether they are positive, negative or neutral perceptions.

- To 'zoom out' to see your project as a whole at a glance
- To demonstrate your findings to others

Working with Model Content

You can represent possible issues or factors in your model using shapes and choose to link these to project items (i.e. sources, tree nodes, relationships) as you create them in your project. On the other hand, you can construct a model using existing items in your project to provide an alternate view of the concepts in your data.

NVivo provides a number of ways to format and work with the content of your models, including:

Groups

Consider using custom and project groups in the following ways:

- To displaying different views on a situation or groups of actors and their relationships
- To represent the perspectives of different team members to discuss different interpretations
- To represent rival explanations in the literature for the phenomena you are studying. Then you can move between these different accounts of the topic area and assess them based on your data
- To break a model into simpler views and bring them together to see a logical whole

In the **Volunteering** Sample Project, the **Perceptions** model contains three different custom groups to gather together items in the model which reflect positive, negative and neutral perceptions. To see only specific types of perceptions, the unwanted custom groups can be hidden.

Styles

In the **Volunteering** Sample Project, a number of model styles have been created. These styles have been applied to items in the **Perceptions** model to add visual emphasis to their membership in the **Positive Perceptions**, **Negative Perceptions** and **Neutral Perceptions** groups.

Adding associations between project items

Consider adding associations between shapes linked to project items for the following analytical purposes:

- To see new insights and recognition of unseen patterns as to how the project items are intertwined.
- To interpret query results. If you add saved query results to your model and add associated items, you will have a visual display of the answer to your query to manipulate and explore the possible explanations.

Dynamic Models

Any shapes you link to project items in your model are 'live' to those items. This means that you can open the item from within the model. Also, if you change an item in any way after it has been linked to a shape in a model, these changes are reflected in the model (i.e. if you change the name of a source which is linked to a shape in a model, the name of the shape in the model will also change).

In the **Volunteering** Sample Project

The **Coding Stage 3** model depicts the wide range of influences on the different motivations the participants had for volunteering and the influences on their perceptions of their own volunteer work.

It is a dynamic model (i.e. it contains links to project items) as its intention is to provide a different view of the information contained within the project.

Static Models

Static models provide a way of showing the development of ideas, concepts and categories throughout your project. You can create a static version of a model which no longer retains links between shapes and any project items they are linked to. Therefore, this static model can show a snap shot of your project at a specific point in time, as the shapes linked to project items will not change.

In the **Volunteering** Sample Project

The **Coding Stage 2** model was constructed following the topic coding of the focus group and interview transcripts. This model was converted to a static model as its intention is to show the understanding of the data at that particular point in the project.

Consider using custom and project groups in the following ways:

- To displaying different views on a situation or groups of actors and their relationships
- To represent the perspectives of different team members to discuss different interpretations
- To represent rival explanations in the literature for the phenomena you are studying. Then you can move between these different accounts of the topic area and assess them based on your data
- To break a model into simpler views and bring them together to see a logical whole

In the **Volunteering** Sample Project, the **Perceptions** model contains three different custom groups to gather together items in the model which reflect positive, negative and neutral perceptions. To see only specific types of perceptions, the unwanted custom groups can be hidden.

Reporting Your Project

Keeping a Project Journal

It is important to account for each significant stage, step and shift in your project and where your ideas came from, as these form the basis of your continued analysis and the conclusions you draw.

Keeping a project journal requires a systematic record of the processes that are significant for your research at the stages that are significant for your project. A document in your project could be dedicated to this process. To keep your journal useful, carefully plan which processes will be logged, and at which stages. (For example, when you finish a stage of observation and commence interviewing, log your reports on nodes and models to ensure that you can clearly tell later what themes were derived from observation.)

Your project journal could include:

- Reports created at various stages in your project to show the progression of ideas, concepts and the state of your data.
- Static models of key concepts and categories at various stages in your project. These will remain as a record after project items or your interpretations are altered.
- Details of the results of the queries you have run at various stages of your project and their contribution to your analysis
- Details of projects which have been merged to comprise the current project

In the **Volunteering** Sample Project

> The **Project Journal** document (in the **Project Notes** folder) contains an account of the project from its inception, descriptions of the data gathered and the growing understanding of the themes at play.

How to Assess My Project?

Qualitative projects require continual assessment of what's known and what's still not understood. They must often adapt to discovery during the project and you continually need to review the project's parts given this change.

NVivo offers the following functionality which you can used to assess the changes in your project in the following ways:

- Visually scan your list of sources to assess richness and relevance of the data
- Visually scan your list of nodes to see how and when your conceptual framework shifted or to see which are most active. (Which nodes are not involved in the current coding—should they be reviewed?)
- Visually scan your queries and results to assess where the activity is, how widely your questioning is ranging, how balanced and strong is the evidence for significant issues.
- Reports allow you to view and regularly store statements of the state of your data, the coverage of your coding, the questions you have asked and results received, the node system and how efficiently it is shaping, the distribution of cases, the balance of relationships and adequacy of evidence for them.

Glossary

A

advanced find: Facility used to find project items based on specific criteria. You access Advanced Find by clicking Options on the Find Bar (at the top of List View.)

ancestor node: Tree nodes or cases above a selected node.

annotation: Text that can be linked to selected content in a source--like scribbled notes in the margin.

attribute: A classification of a case, such as gender, age or location

attribute value: The values of an attribute. For example, 'male' or 'female'.

audio: Source materials such as recorded interviews, music, sound effects, and other forms of audio that may be relevant to your research. Types of audio files that can be imported into NVivo include *.mp3, *.wma,*.wav. An audio source contains the audio file and a transcript column.

auto code: A quick way of coding that uses heading styles or paragraph numbers to create nodes and code at them.

B

boolean operator: The use of AND, OR or NOT to combine search terms.

C

case: A node with attributes such as gender or age. You can use cases to gather content about a person, site, institution or other entity involved in your research. Like tree nodes, case nodes can also be organized in hierarchies.

casebook: A matrix displaying cases, attributes and attribute values. You can create cases, attributes and values in NVivo or you can import them from a tab-separated text file. To open the casebook, on the Tools menu click Casebook > Open Casebook.

child node: A node below a parent node.

classification: In NVivo, 'classification' refers to relationship types and attributes. Relationship types provide a way of 'classifying' relationships and attributes provide a way of 'classifying' cases.

coded at: When you select text and categorize it as belonging to a specific node (theme or idea), the text is said to be 'coded at' the node.

coding: Selecting source content and defining it as belonging to a particular topic or theme. By creating nodes and coding at them, you can catalogue your ideas and gather material by topic.

coding context: The words, paragraphs and heading levels that surround coded text in a source. When exploring a node, you can choose to spread coding to the selected context.

coding density: Areas in a source or node in which most coding occurs. The Coding Density bar is visible when you display Coding Stripes. The color graduations indicate the coding density from light gray (minimal coding) to dark gray (maximum coding).

coding excerpt: A passage of text coded at a node. When exploring a node, you can set display options for coding excerpts (View>Coding Excerpts).

coding reference: An occurrence of coding. When you open a node, you can see all the references to source material that are gathered there.

coding stripes: Colored stripes that enable you to see coding in a source or node.

connector: A line that joins shapes in a model.

coverage: The percentage of a source that is coded at a node.

custom group: A model group that you create. You can include shapes and connectors in the group and show or hide them as required.

D

detail view: The bottom-right pane in NVivo. You explore documents, nodes and models in this view. You can choose to 'undock' detail view if you want to work with sources, nodes or models in a separate window.

document: Source material such as field notes, transcripts, interviews, literature reviews or whatever material that is relevant to your project. You can 'code' a document (or any part of it) to categorize the information that it contains. You can import documents or create them in NVivo.

E

embed: To store a media file inside your NVivo project as opposed to linking to a file stored externally.

external: Source material that cannot be imported into NVivo. This might include items such as newspaper articles, books, video footage or audio tape. You can use the external to represent the un-importable material and record any notes or summaries that can be coded as required.

F

folder: A place in Navigation View for storing your project items. You can create your own folders for organizing sources, queries and models.

free node: A free node is a 'stand-alone' node that has no clear logical connection with other nodes and does not easily fit into a hierarchical structure. You can convert a free node into a tree node by moving it into a tree node folder.

G

grouped find: Grouped Find enables you to list selected items and find the items related to them. You access Group Find by clicking Options on the Find Bar (at the top of List View.)

H

hyperlinks: A link from content in a source to a file or URL outside of your NVivo project.

I

image: Term is used to refer to graphic or photo files found outside a picture source. Types of image files include bmp, gif, jpg, jpeg, tif, and tiff.

L

links: In NVivo, links refer to memo links, annotations and 'See Also' links.

list view: The top-right pane in the NVivo window. You view the contents of your NVivo folders in List View.

log entry: Comments, descriptions, notes, hyperlinks, or ideas entered against the whole or portion of the image in a picture source. A picture source may or may not contain a log entry.

M

matrix: A matrix is a collection of nodes resulting from a Matrix Coding Query.

media file: Refers to both audio and video files.

memo: A type of source that you might use to record thoughts and observations. If a memo is related to a particular source or node you can create a 'memo link' and link the two together.

memo link: The link between a source or node and a memo. A memo can only be linked to one item.

mixed method: The combination of quantitative and qualitative research methods.

model: A visual representation of your project and its contents.

model style: A set of formatting characteristics that you can apply to shapes or connectors in a model to quickly change its appearance.

N

navigation view: The panel on the left side of the NVivo window. It contains buttons that enable you to access project items.

nickname: A short name given to nodes for quick coding.

node: A container for a theme or topic within your data. For example, you can create a node called 'community' and code all community-related data at it. When you open the node you can see all the community-related data gathered in one place. Types of nodes include, free nodes, tree nodes, cases, relationships, matrices and results.

P

paragraph: Text or images between two carriage returns. You can apply a style to content in a paragraph.

paragraph number: Paragraph numbers can be included when printing or exporting a source or node. In a node, paragraph numbers relate to a reference's position in the source.

parent node: A top tree node or case which is above other nodes in a hierarchy.

picture: A type of source that contains a picture file and log entries.

picture file: Can be an image copied from a document, a frame from a video source, or a region from a picture source. Types of pictures files include bmp, gif, jpg, jpeg, tif, and tiff.

playhead: Refers to the blue slider that indicates the point where the playing/paused media is at. It can be dragged to allow play/pause from any specific point in a video or audio file.

project group: A group within a model that is created based on existing attributes or relationship types. These groups enable you to show or hide cases based on their attribute values and relationships based on their type. You cannot add items to or delete items from project groups. To use these groups, your model must contain cases or relationships.

Q

query: A way of asking questions about your data. You can save a query and run it as your project progresses.

R

read-only: Source content that cannot be edited. A source is read-only when coding stripes are displayed or when the read-only check box is selected in the source's properties.

region: A selected portion of a picture.

relationship: A node that defines the connection between two project items. For example, the relationship between two cases (Anne loves Bill) or between two nodes (Poverty impacts Health).

relationship type: A word or words (usually verbs) which define the relationship between two project items. For example, 'impacts', 'causes' 'employs' 'loves' and so on. Relationship types also have a direction.

relevance: In text search results, relevance indicates the fins which are the 'best match' for the scope and criteria you have defined. The rating is derived from (1) the relevance weighting given to the text (if any), (2) the number of scope items, (3) the number of finds in the scope, (4) number of finds in a particular scope item and (5) criteria in your query.

results: A node or list of project items resulting from a query. You can store a results node in the Queries Results folder or move to the main node system for coding.

right-click: Click the button on the right side of your mouse.

S

see also link: A link from selected content in a source or node to selected content (or entire content) in another source or node.

set: A collection of shortcuts to project items.

shadow coding: Indirect coding in an audio or video source - when you code a transcript entry, the corresponding portion of the media is 'shadow coded'. When you view coding stripes, this 'indirect' coding appears as a shadow on a coding stripe.

sibling node: Tree nodes or cases that share the same parent node.

source: In NVivo, 'sources' is the collective term for your research materials anything from hand-written diaries to interview transcripts in Microsoft Word format. You store sources in the Documents, Externals or Memos folders.

static model: A 'snapshot' of an existing model. You cannot edit static models and they are not linked to live data.

system folder: Folders that are supplied with NVivo such as documents, memos and externals. You cannot delete or rename system folders.

T

text style: A set of formatting characteristics that you can apply to text to quickly change its appearance.

thumbnails: Miniature images or graphics. They refer to List View options which allow you to display items in small, medium or large sized miniature graphics for easy identification and review.

timeline: Displays the duration of the audio or video file.

timespan: A timespan is the duration of time for a transcript entry. For example, Jane spoke from the two minute point to the ten minute point (00:02:00-00:10:00). When importing transcript entries from a table, you can include a timespan for each row in the table.

transcript: Contains audio or video transcriptions against specific timespans. You can also include notes, hyperlinks or comments in the transcript or content column as needed in your research. Transcripts can be coded on their own or as part of a specific timespan.

tree node: Nodes that are organized in a hierarchical structure moving from a general category at the top (the parent node) to more specific categories (child nodes). You can use them to organize nodes for easy access, like a library catalogue.

U

user profile: Includes the name and initials of a user or team member.

V

video: Source materials such as focus group discussion videos, tv ads and other forms of video that may be relevant to your research. Types of video files that can be imported into NVivo include *.mpg, *.mpeg,*.wmv, *.avi, and *.mov. A video source contains the video file and a transcript column.

video frame: A static picture that can be captured from a video.

W

wildcards: A keyboard character such as an asterisk (*) or a question mark (?) that is used to represent one or more characters when you are searching for project items such as sources, nodes or sets. For example: g*t will find get, great and gt