Qualrus
Distinguishing features and functions

This document is intended to be read in conjunction with the 'Choosing a CAQDAS Package Working Paper' which provides a more general commentary of common CAQDAS functionality. This document does not provide an exhaustive account of all the features and functions provided by Qualrus but is designed to highlight some of its distinguishing elements. The Comment section at the end details our opinions on certain aspects of functionality and usability. See also Lewins & Silver (2007) *Using Software in Qualitative Research: A Step-by-Step Guide*, Sage Publications and software developer website.

Background http://www.qualrus.com

Qualrus is developed by Idea Works, Inc. with developmental support from qualitative researchers including Howard Becker. It differs significantly from other CAQDAS packages because it uses 'intelligent computational strategies' to assist the researcher with certain qualitative tasks. It may therefore be seen as the first in a new generation of qualitative data analysis software packages.

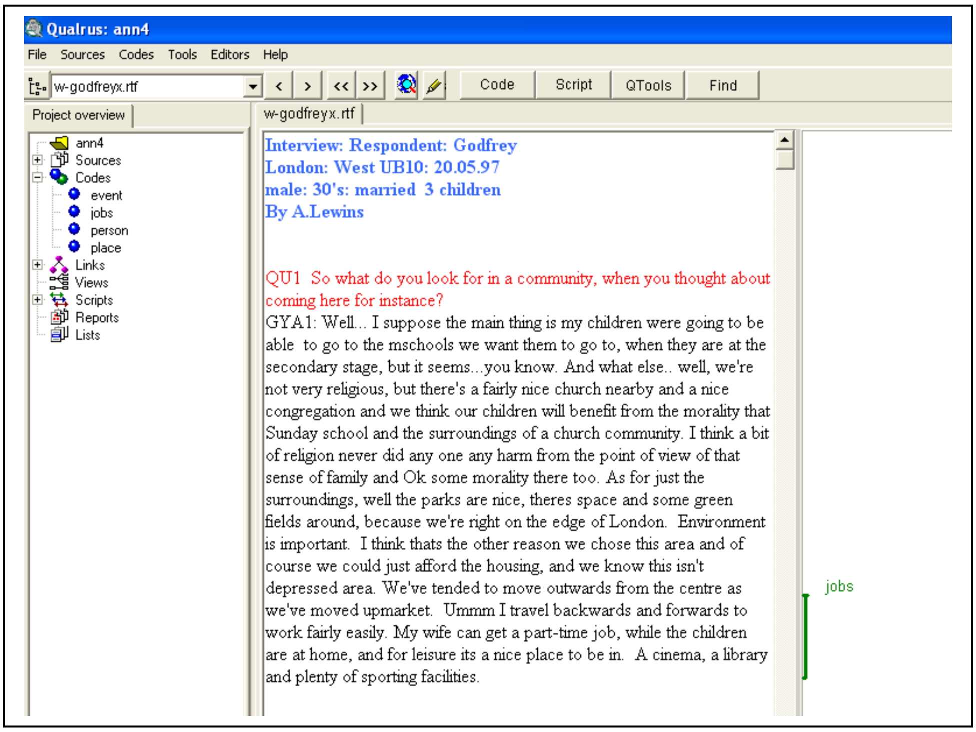
Minimum System Specifications (recommended by developer)

MS Windows 95 or higher ■ 48Mb RAM (required), 128Mb (recommended)

Structure of work in Qualrus

Qualrus functions using an external database structure whereby data files are read rather than contained by the software. ■ The ' pack and go' system for condensing and backing up is efficient and easy to use. ■ The Qualrus interface comprises three main parts: The Project Overview contains links to project items (sources, codes, links, views, scripts, reports and lists); the Sources Window displays data files; and the Margin View displays assigned codes. ■ Other functions are accessed through main menu options and short cut icons. ■ The segment is the basic unit of analysis and is created upon coding. This affects the way certain search tools function and it is important to be aware of this structure early on.

Figure 1. Qualrus Interface



Interview: Respondent: Godfrey
London: West UB10: 20.05.97
male: 30's: married 3 children
By A.Lewins

QU1 So what do you look for in a community, when you thought about coming here for instance?

GYA1: Well... I suppose the main thing is my children were going to be able to go to the mschools we want them to go to, when they are at the secondary stage, but it seems...you know. And what else.. well, we're not very religious, but there's a fairly nice church nearby and a nice congregation and we think our children will benefit from the morality that Sunday school and the surroundings of a church community. I think a bit of religion never did any one any harm from the point of view of that sense of family and Ok some morality there too. As for just the surroundings, well the parks are nice, theres space and some green fields around, because we're right on the edge of London. Environment is important. I think thats the other reason we chose this area and of course we could just afford the housing, and we know this isn't depressed area. We've tended to move outwards from the centre as we've moved upmarket. Ummm I travel backwards and forwards to work fairly easily. My wife can get a part-time job, while the children are at home, and for leisure its a nice place to be in. A cinema, a library and plenty of sporting facilities.

Data types and format in Qualrus

Textual formats : *Text only & Rich Text Format.* ■ Graphic formats : *.bmp, *.jpg. ■ Video formats : *.avi, *.mpeg, *.mpg etc. ■ Audio formats : *.wav, *mp3 etc. ■ Html formats : *.htm, *html etc. It is also possible to navigate the internet from within Qualrus and save pages as sources. ■ See developer website for complete list of acceptable source file formats.

Closeness to data and interactivity in Qualrus

Data are viewed in the Sources window and loaded sources are tabbed for quick access. ■ There is usually good interaction between different windows although limited drag and drop functionality. ■ The margin view is interactive, providing quick access to memos, the coding schema and other project items.

Coding scheme in Qualrus

The coding scheme can be as hierarchical or un-hierarchical as required. Hierarchy is imposed by linking codes in a semantic network rather than creating sub-codes. ■ Code labels can be given synonyms and attribute-value pairs (e.g. Gender-Female) can be applied to codes where relevant. ■ Summary information concerning each code is held in the Code Editor window, which also provides quick access to coded segments and visualisation of links between codes. ■ The approach to coding and the structure of the coding schema can depend on whether and how suggestive coding functions (see below) are made use of.

Coding processes in Qualrus

Suggestive Coding if utilised, provides an active role in coding processes. Codes are suggested for application to data segments based on the combination of natural language patterns; user-defined code synonyms and links between codes. For example, codes can be suggested by the program on the basis of comparing data segments according to text patterns and previous code applications. The software 'learns' as coding proceeds, based on the acceptance or rejection of previous suggestions. ■ Suggestive coding features work incrementally; codes are initially suggested upon text selection because a) the code label itself or a synonym (or its root) appears in the selected text, or b) because a code has been applied earlier in the document (constant comparison). Upon applying a code (x) to a text segment (whether suggested by the program or not), further codes may be suggested because a) code 'y' has previously also been applied to text segments coded by 'x'; or because b) a one-directional link exists between code 'x' and code 'y'. ■ The Coding Form allows application of suggested codes or any other codes. Codes can only be applied to segments, but a segment is defined by the user and can be any size.

Basic retrieval of coded data in Qualrus

Retrieval of coded data happens via the Code Editor or by using the search tools (QTools). In the first instance segments are 'lifted-out' of context but there is good interactivity between results pane and source text.

Data organisation in Qualrus

Organisation of data to known characteristics such as socio-demographic variables can be achieved by coding or by assigning attribute-value pairs to codes.

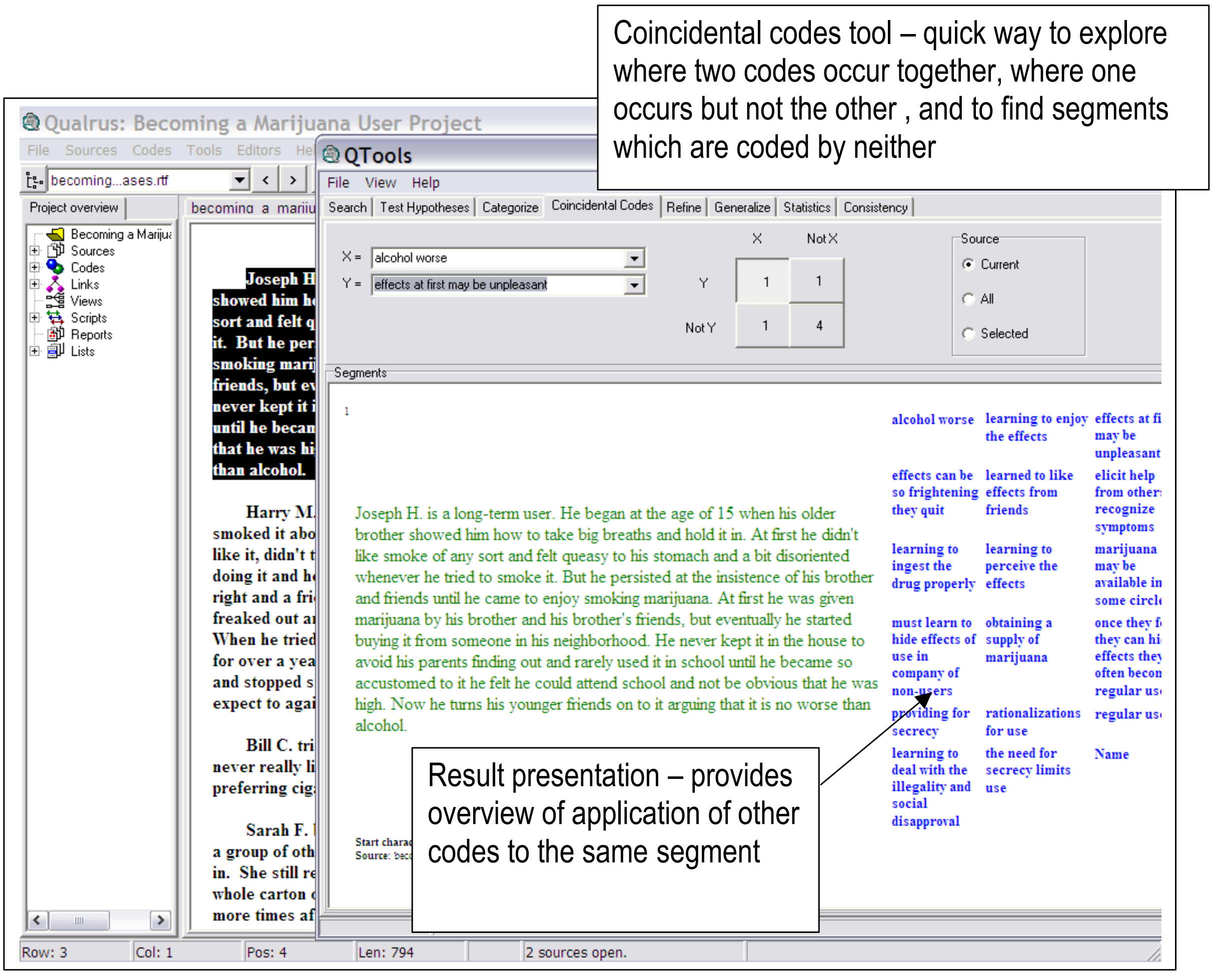
Writing tools in Qualrus

Memos can be attached to sources, codes and segments although there is no central memo system where all memos are listed. Empty sources can be created as memos and worked with like any other source. QTools allow searching for coded segments with an attached memo.

Searching and interrogating the dataset in Qualrus

Qualrus provides a number interrogation tools, accessed through QTools, in addition to standard Boolean operators. Much information is automatically generated without having to construct complicated query expressions. ■ Good interactivity exists between QTools results and source data. ■ The Statistics Tool assists with theoretical sampling, providing basic summary statistics (e.g. frequency of code application per source and frequency of code co-occurrence at a particular coded text segment). ■ The coincidental codes tool examines the co-occurrence of each possible pair of codes in segments, ordering pairs by how frequently they co-occur. ■ The categorizing tool allows coded text segments to be sorted into stacks which themselves can be assigned codes. This is particularly useful for generating higher order categories outside of the main coding schema, and reflects manual methods of working. ■ The hypothesis testing tool examines all segments to test specific user-defined hypotheses. For example, search for where two or more codes are applied to segments and specify for another code to be assigned to all finds, using standard IF, THEN protocols. All possible combinations of the codes asked for are presented in a matrix result which facilitates the 'finding' of where co-occurrences are not present in the data, as well as where they are. ■ The refine tool examines codes or relationships among codes and can be used effectively to build on work done in the hypothesis testing tool. ■ Auto-coding tools are based on the use of a script that first assigns a dummy code to a segment. The software then checks to see if it should be coded with the auto-code specifications, then uncodes it if it is still coded with the dummy code. ■ Lists save results from which to build further questions upon.

Figure 2. QTools



Coincidental codes tool – quick way to explore where two codes occur together, where one occurs but not the other , and to find segments which are coded by neither

X = alcohol worse
Y = effects at first may be unpleasant

	X	Not X
Y	1	1
Not Y	1	4

Source
● Current
○ All
○ Selected

1

Joseph H. is a long-term user. He began at the age of 15 when his older brother showed him how to take big breaths and hold it in. At first he didn't like smoke of any sort and felt queasy to his stomach and a bit disoriented whenever he tried to smoke it. But he persisted at the insistence of his brother and friends until he came to enjoy smoking marijuana. At first he was given marijuana by his brother and his brother's friends, but eventually he started buying it from someone in his neighborhood. He never kept it in the house to avoid his parents finding out and rarely used it in school until he became so accustomed to it he felt he could attend school and not be obvious that he was high. Now he turns his younger friends on to it arguing that it is no worse than alcohol.

alcohol worse
learning to enjoy the effects
effects at fi may be unpleasant
effects can be so frightening they quit
learned to like effects from friends
elicit help from other recognize symptoms
learning to ingest the drug properly
learning to perceive the effects
marijuana may be available in some circl
must learn to hide effects of use in company of non-users
obtaining a supply of marijuana
once they f they can hi effects they often becon regular us
providing for secrecy
rationalizations for use
regular us
learning to deal with the illegality and social disapproval
the need for secrecy limits use
Name

Row: 3 Col: 1 Pos: 4 Len: 794 2 sources open.

Result presentation – provides overview of application of other codes to the same segment

Linking devices in Qualrus

Codes can be linked to one another using default or user-defined links. Default links are 'causes', 'excludes' and 'is a'. ■ Suggestive coding is partly reliant on links specified between codes, although this feature can be disabled if required. ■ Links between codes can be created at any time and are centrally visualized in the Code Editor. ■ Specific parts of the semantic network can be further interrogated and visualized in maps by working in different View Editors (see below). Data segments cannot be linked independently.

Mapping in Qualrus

Creating links between codes can be a central aspect of work in Qualrus when making use of suggestive coding functionality. ■ The direct links from one code are shown in the Code Editor, but the View Editor provides ways to focus in on parts of the coding schema to visualize and manage links in different ways. ■ The Network View enables the visualization of any number of code links. This facilitates the comparison of similar codes and the visualization of hierarchy. ■ Certain types of links (e.g. 'causes' or 'is a property of' etc.) can be viewed in isolation from other link types in the Network View. Only codes that exist within the project can be added to Views; it is not possible to add non project items such as graphics, or use colour or layers to group different aspects. ■ Codes in a view are not interactively linked to the rest of the project. ■ Views can be saved as pictures (.bmp file).

Output in Qualrus

There are many different ways to generate output concerning aspects of work in Qualrus. Output options allow the content and format of output to be specified in various ways. ■ Reports save output for viewing in other applications. Reports can be saved in basic or dynamic html format.

Extensibility in Qualrus

Qualrus is extensible in that the program can be modified to fit particular needs. ■ Write scripts using an object-oriented scripting language (similar to C++ or Java) to extend and tailor the software.

Team-working in Qualrus

The merge facility allows all or selected aspects of one project to be imported into another. The coder consistency tool facilitates team-working by providing statistical data showing how consistent each coder is with the recommendations of the program. ■ The software automatically logs all aspects of work, thereby providing information concerning, for example, the details of a code (it's name, list of segments applied to the code, when the code was created, by whom and how it was chosen – i.e. whether selected manually by the user, or as the result of a program suggestion).

Comment on Qualrus:

Qualrus is a sophisticated software *in some unique respects, although it can be used effectively without making extensive use of sophisticated tools. It is easy to get started with, although takes time to become familiar with the range of options available. Several functions are more quantitative than is usual amongst CAQDAS packages, providing alternative ways to compare, for example, the application of codes statistically.*

Use of artificial intelligence *in particular, takes qualitative software to a different level while also being a controversial aspect amongst some users. The expert user may make good use of the AI features, although the functioning of this aspect of the software needs to be well understood early in order to make effective and appropriate use of available tools.*

Suggestive coding *may be seen as a way to speed up coding processes, but more importantly, it can serve to increase rigour and consistency of coding. Making full use of suggestive coding is facilitated by developing a* *fairly mature coding schema at the outset, and may therefore be particularly useful to those working in deductive ways. That said, Qualrus is equally appropriate for more inductive approaches to analysis, and the nature of its AI constant comparison with segments coded before has some considerable resonance with Grounded Theory. (Glaser and Strauss 1967)*

The segment structure is a little restrictive *in comparison to other CAQDAS packages, specifically the current inability to search for where codes merely overlap in the data and are not applied to exactly the same segment. There is not yet a straightforward way to compare codes across subsets of data.*

Interactivity *in most windows is good with other aspects of work, although some dialogue boxes, including the Coding Editor, are somewhat 'clunky'.*

Extensibility *means that some of the program's limitations can be overcome by the confident computer user willing to learn the scripting language. Where this is the case, its potential is almost unlimited.*

The Categorisation tool *replicates manual ways of grouping text segments and is very intuitive. This feature can be particularly attractive to experienced qualitative researchers who are familiar with manual methods of coding and categorisation and therefore seek software which replicates that style of working.*

Data and project management *options are not as wide-ranging or straight-forward as other packages.*

Further Reading

- Brent, E., & Slusarz, P. (2003) "Feeling the Beat" : Intelligent Coding Advice from Metaknowledge in Qualitative Research, *Social Science Computer Review 21(3) 281-303*
- Gibbs, G. (2007) **Analysing Qualitative Data**, part of the Qualitative Research Kit, ed. U. Flick, Sage, London
- di Gregorio, S & Davidson J (2008) **Qualitative Research for Software Users**, McGraw Hill, Open University Press, UK
- Lewins, A. (2008) 'CAQDAS: Computer Assisted Qualitative Data Analysis' in (ed) N. Gilbert, **Researching Social Life** (3rd ed), Sage, London
- Lewins & Silver (2007) *Using Software in Qualitative Research: A Step-by-Step Guide*, Sage Publications (uses an NVivo project as worked example)
- Silver, C., & Fielding, N. (2008) 'Using Computer Packages in Qualitative Research', in Willig C & Stainton-Rogers W (eds.) **The Sage Handbook of Qualitative Research in Psychology**, London, Sage Publication