

Working paper.

Qualitative phase of the formative evaluation of learning training needs in computer assisted qualitative data analysis.

Graham R. Gibbs, University of Huddersfield
Celia Taylor, University of Huddersfield
Professor Nigel Fielding, University of Surrey
Ann Lewins, University of Surrey

Online Support for QDA and CAQDAS and Evaluation of Learning Needs. RES-333-25-0009

THE RESEARCH PROJECT

The formative element of evaluation of the Online QDA and CAQDAS project included a qualitative phase, comprising data collected in different ways and from users with different levels of experience. This interim working report summarises our progress with the qualitative element of evaluation, and refers to areas still awaiting completion. The data were analysed with the assistance of QSR NVivo.

RESEARCHERS AND DATA COLLECTION

Lewins observed ten volunteers working with a range of CAQDAS programs while they talked about their work to date in their current qualitative project and asked for advice where they felt they needed it. Taylor interviewed 24 software users and trainers, in semi structured interview formats, either face to face or over the phone. All sessions were to be transcribed. This process is still being completed and work will be ongoing for the purpose of further reports and papers. They also used informal help-line telephone/email data over a 3-month period, collected by Lewins in her role as manager of the CAQDAS Networking Project. Reporting from the qualitative analysis phase has already informed some parts of the web resource on QDA and CAQDAS which has been constructed as part of the project (onlineqda.hud.ac.uk).

Table 1. Types of respondents taking part in the interviews and help/observation sessions

Type of respondent	Numbers
Researcher	11
Software trainer	3
PhD student	9
Professor	1
Lecturer	5
Qualitative IT managers	2
Qualitative resource manager /s/w trainer	2
Masters student	1
<i>Total</i>	34

The sample was designed to include a range of the different kinds of users with differing levels of experience and expertise, and with varying roles – PhD student, software trainer, researcher etc.) See Table 1 and Table 2.

Due to time constraints, some of the later sessions have yet to be transcribed (7) but the transcription is being completed and will be incorporated in analysis featured in future papers

Table 2. Discipline of respondents (transcribed so far)

Discipline	Respondents
Nursing	1
Psychology	4
Education	4
Sports science	1
Govt Audit research	1
Pharmacy	1
Health	2
Public health	1
Men's Health	1
Sociology	3
Historic Anthropology	1
Built Environment	1
Psychiatry	2
Management	3
Qual. resource	1
Trainer-Consultant	1

THE TEAM DIMENSION

Preparing to work as a team, especially when members are geographically remote and especially where the team plans to 'merge' separate inputs from each researcher is a problematic part of the project design and project management. Lewins and Taylor made a number of simple decisions about the division of work, and intended to use the processes of managing the team dimension and any problems they met as material to inform the building of protocols for the Online QDA and CAQDAS web resource. For later publications, Lewins and Taylor are compiling a plain English table in which they retrospectively and chronologically summarise issues, happenings, irritations

and finally commentary on solutions, better practice and recommendations. It will form the basis of a later paper, and has already informed parts of the web resource.

THE INTERVIEW FRAMEWORK

The aim of the interviews was to investigate current use of CAQDAS software by a range of users (see Table 1). The interview framework was designed to explore what users' experiences had been whilst analyzing qualitative data in CAQDAS packages. Researchers, software trainers, PhD students, lecturers, qualitative IT managers, and resource managers were interviewed. The factors identified during an initial analysis of the software discussion forums helped construct the interview framework. The interviewees were informed that the purpose of the survey was to collect information to help design a web based resource for QDA and CAQDAS. Most informants took part in a telephone interview which lasted approximately 45 minutes. Interviewees gave details of their current software use, experiences, difficult areas and what they would like to see included in the web resource. The interview framework covered the role of respondents, current project details, the training and support they had received, the software-learning experience, and their own evaluation of what was needed in an online provision. For trainers the framework was modified to focus more on their experiences of teaching the software and supporting users during their various project lifetimes. The interview guidelines were also used to supplement information gathering during the help/observation sessions.

THE HELP/OBSERVATION SESSIONS

Eight observation sessions took place in which researchers talked about and demonstrated their own projects. They demonstrated how they were using CAQDAS software and discussed aspects of software use and analysis and where they needed assistance. The more traditional applications of CAQDAS software were represented, for instance, projects concerning access to health care amongst marginalized groups, psychiatric research on self harm amongst adolescents, the introduction of training schemes for bipolar disorder and education related projects. The built environment (a discipline increasingly seeking training in CAQDAS) was represented by a project on safety in the home. More innovatively, a historical documentary analysis of Elizabethan letters in English and Latin, concerning the health issues of the Cecil family became a project supported at several stages of its development. This small group was also indicatively representative in terms of previous support received; one user was self-taught, several had received previous CAQDAS training, and at least two had no background in qualitative methods training.

MAIN THEMES

The interview framework itself worked at a practical level to inform the building of the online resource. However, we began to see other themes emerge in terms of how each respondent talked generally about the place of software in their own and their peers' work and what we perceived about the way they talked of and performed qualitative research. Some of the themes were personally oriented, others by necessity were about practical issues to do with day to day work with software. This report concentrates on the most important topics in terms of influences on how the resource was constructed; these were teaching and learning issues, analysis, the effect of software on method, practical difficulties and insecurities. See Table 3

Table 3. Summary of themes covered in report

Analysis	
Software's influence on	-Expressions that once comfortable with software users tended to make

method	do with familiar processes – without stretching. - Cautiousness about the persuasive ease of some software functions. - In applied research, occasional reliance on autocoding over more careful work. -Resistance to forceful software developer statements about method.
Teaching QDA and CAQDAS	-Variable local support and investment -QDA teaching complicated by CAQDAS -Need for 'joined-up' methods teaching and avoidance of 'easy option' perceptions amongst students
Learning issues	-Lack of local support and so failure to sustain learning and familiarization momentum after initial training
Limited perceptions of software and qualitative data analysis	-Often more to do with misunderstanding the relationship between software and qualitative data analysis (hoping for magic outcome) -Related to teaching issues, software and qualitative data analysis seen as easy option by undergraduate students
Self reliance	-Several users liked to work things out on their own – however, at least two self taught researchers seemed to have problems perceiving the larger picture of what the software could and couldn't do, or had limited or unrealistic perceptions of its usefulness.
Changing expectations and attitudes to software	- Recognition of the necessity of CAQDAS teaching but availability of software sends wrong messages to inexperienced QDA students etc. - Software has complicated qualitative methods teaching.
Difficulties in use of software	Practical issues: - Most prevalent were problems with Searching. -Muddles about where coding was happening in complex hierarchies - Too many ways to do one thing. -Bringing different aspects together e.g. socio demographic variables and conceptual coding
Fears - insecurities	Different from difficulties, much more related to uncertainty about quality of work and twofold anxieties about how muddled /inefficient/ use of software caused wrong turns in thinking. Also prominent security issues about saving and moving projects

TEACHING QDA AND CAQDAS

Amongst the 21 respondents, 8 were trainers of QDA or software, some of them also users. Of those, 3 were teachers of a range of packages and were specialists in all CAQDAS software, 4 concentrated on one package and their role was to teach QDA and CAQDAS software as part of a curriculum, 1 was a private consultant/trainer specialising in one software package.

Amongst teachers involved in the teaching of students at HE institution, the place of CAQDAS in the curriculum was either as part of an expectation that the students were about to use the software, or was part of an awareness raising exercise. The ways in which they talked about software teaching and especially support clearly reflects the local resources invested in this aspect. Heavy investment in two institutions in *locally* based computer specialist staff revealed in one case intensive input and support for one package, and in the other a wider ranging, less intensive support for more methodological (qualitative) starting points and more packages. Other trainers reported that universities regularly bought in externally sourced software tutors in answer to demand for campus-wide provision, especially where IT services were supporting provision. Other institutions arranged this on a much more local departmental or even project-oriented basis.

It was unsurprising that the two locally funded qualitative IT managers who offered support to CAQDAS users, were not particularly vocal about the teaching of QDA itself or the relative position of CAQDAS within the spectrum of research methods teaching. By contrast those who

were involved in both elements (QDA and CAQDAS) had much to say on the subject. Expectations of qualitative data analysis and how CAQDAS contributes to them were the concerns of one senior academic. She did not count herself as an expert in CAQDAS and her current efforts were going into creating a more coherent research methods environment for undergraduate and postgraduate students,

"I actually taught them how to use software... must say basically I don't know how to use it myself. And I think also that, there is an idea from a lot of students, that software is a kind of means to end as opposed to a tool to try and get there. They think you can put the data into the software then it's going to do the analysis for you, which in fact it doesn't, does it? It's a tool for helping you too look at the data as opposed to doing the actual cognitive thinking... ..what might be in the data".

It appears that there is some way to go before CAQDAS becomes embedded in the curriculum in the same way as statistical software; there were only 4 institutions represented in the sample where there appeared to be much more than awareness raising support for CAQDAS. Though not a 'user', in her teaching she spoke of the utility and limits of the CAQDAS tools and worried about the undercutting of thinking skills and the principles of good qualitative analysis. Too much concern on the emphasis on learning software, using the tool, cut away at the time given to learning how to look deeper and to think more analytically. Later in the same interview, the respondent portrayed a prevalent misconception about qualitative data analysis amongst students;

"Because I think, especially the first years, they get some easy options, instead of doing SPSS, you know they're scared stiff of numbers, so they think 'oh I'll do something qualitative because it's easy' and sort of getting across to them that it's not easy, it's intellectually demanding and you have to be very critical of yourself and also of the data that's the end product that you're producing. So it's getting past that kind of notion that it's kind of soft option if you like".

She went on to raise a much voiced concern

"...and also moving from that kind of descriptive phase of analysis to the sort of analytic phase of analysis I think that's quite hard. I think, it's quite hard when you first start. What I try to say is it takes practice and it takes time".

This reflected a view from students that the added dimension of software made the easy option seem even easier, yet in reality for this course director it made the teaching of QDA more problematic. One of the qualitative IT specialists in another institution, providing local training and support for one package only, touched on the appeal and dangers of software use

"..... and they all love auto coding...and think that will do the work for them quickly, so they get a very firm talking to about that".

This clearly resonates with issues discussed later with occasional users who start out with very limited perceptions both about what QDA itself is, and linked to that, how CAQDAS is supposed to 'do it for you'.

Concerns about 'doing the analysis' are interrelated with teaching issues. Another respondent, an experienced software user and trainer, commented on a researcher at a later stage of the analytic process, who attended training. The researcher had already thoroughly coded her dataset (self taught in NUD*IST-N6) on an intervention for mental health clients;

"she was very quick and very able and she sort of looked at me at the end (of the project specific help session) and she said 'So... it's not going to do, there's no magic thing that, that's going to do it for me then?' and she really had come along hoping that there was

some...something that the software could do that was actually going to pull out the analysis for her”.

It transpired later that this had not been just her own view but remained for some time as a pressure from her quantitatively experienced supervisor, who insisted that there must be analytic outcomes from the software which had been missed.

Another trainer, an experienced user of NVivo, suggested that the traditional craft skills are a necessary base from which to move to software use;

“...And I also stipulate that if you’ve got say 4 or 5 interviews it’s much easier to do it by hand. And actually when I teach the students I teach them how to do it by hand because I think there is a...it goes back to that feel of picking up the papers, reading it, marking it, cutting it, they have to know what that means before they can really transfer that skill on to Nud.ist”

In evaluating general requirements of curriculum support for QDA and CAQDAS combined, as the senior academic stated, the process needs to be coherent;

“...we should be teaching them philosophy of science and reflexivity before they actually do the qualitative data analysis. And that happens in undergraduates and in the sort of postgraduates as wellIt’s just a matter of it being joined up properly”

Though we did not ask the question specifically, such views are implicitly related to ideas about the stage at which software itself should be taught and in particular the suitability of the CAQDAS software dimension being included at undergraduate level. Another in-house trainer (NVivo only) said,

“...there’s all sorts of modules available for people to go and learn how to do Qualitative Research in, but there’s only these Workshops where they can work out, they can learn how to use the package. We don’t open it up to Undergraduates. My argument is the learning you need to be able to use it for a small scale study isn’t worth it. It’s a researchers that takes ... you need a tool, if you’re either employed as a researcher or doing a post graduate research project, that’s when it’s worthwhile.”

LEARNING NEEDS

In exploring related *Learning* concerns amongst users (not trainers or methods lecturers) little was explicitly acknowledged about the need for more qualitative data analysis assistance, though several were explicit or less explicitly revealing about a lack of awareness in this respect. A PhD student and MAXqda user (Phys Ed Sciences) admitted that in the absence of any other real knowledge about methodology, he used the software to provide a working structure to his qualitative work. An N6 student user (Mental Health), though very aware about her needs and capable with the software, had unconsciously betrayed a lack of awareness about the relationship between software and analysis, which was much to do with her more quantitative background and the different relationship there between software and analysis. It was clear that PhD students do not always get the right, targeted support for their specific project. Helpline calls and mentions in our interview data supported the need for exemplar projects that linked work done in a project with software processes –

“but there isn’t really a lot written on here’s my methodology and here’s it working within the software and here’s a data base you could look at to get ideas and ways to work it. So I find it quite tricky to be honest”.

Another PhD student user suggests -

“...maybe discipline specific uses of qualitative software analysis. That would be fantastic.”

Most of the explicit talk was oriented to the need for ongoing learning and support needs with software *after* initial training when using the software for their own work. There were several comments about the lack of local ongoing support for the use of software. Even those who seemed fairly self-reliant in terms of teaching themselves a program could see they had some problems in using it effectively

"I'm probably like a lot of other people who are experimenting with computer assisted QDA that I'm happy enough with technology to figure out the nuts and bolts of how the program works. You know I can figure out a new software package given a bit of time and help money, that's no problem, it's the how do I use it?"

Several others, however, expressed difficulties in learning from software manuals sometimes because of the way they are written or impatience on the part of the reader; additionally this quote was typical of several users who learn through more interactive processes -

"I'm one of those people that learns better from doing and talking, conversing. I'm not very good at learning from, you know, picking things up from manuals or from the help systems within pieces of software".

Even here, however, different opinions about manuals were closely connected to degrees of personal ease with software, and in two cases at least, researchers were complimentary about official software help.

"A demo exactly and it has a manual with it so I kind of used...so I guess that is training in a sense I used one...I found it...I didn't find it very good [NVivo demo tutorials]. So that was why it put me off using it.. I did the same with MAXqda. I used the MAXqda manual and I managed to find it very supportive."

Whereas, the researcher concerned with the analysis of historical Elizabethan documentation (and very new to the concepts of CAQDAS) found the supplementary literature on NVivo a great help –

".. well I bought this book that you recommended [Gibbs 2002-Explorations with NVivo] and I feel that if you had some in the bookshop for people to buy so that you'd say go home with this book, it's really good, and I annotated when I find things and I read bits of it.... that's very useful and very practical and I used this one that came with the thing and I annotate that as well [in-software NVivo tutorial] so that's what I've done mostly but I once e-mailed you in a state and then once I came to see you..."

For others there were basic computer skills which needed to be addressed to increase readiness to take on board new information. Particular needs are referenced in the next section. A trainer of several software programs highlighted the difficulty of getting it right for individuals where there was such a wide variations in skills and knowledge -

"...I've had people on my course that have barely known how to use a mouse to people who can sit down very rapidly, see the potential, move on and there's people that are struggling with QDA and others who are very au fait with Qualitative Data Analysis but have never used the electronic source".

FEARS AND INSECURITIES

In discussing problems concerned with actual analysis and usage of software we chose to make a distinction between what have been described here as 'fears and insecurities' and on the other hand 'difficulties'. Main difficulties in general (itemized in Table 3), were less emotively referred to and more practically based whereas fears and insecurities were quite deep-seated concerns not only about safety but the value of the work they were doing. For new users there was a fear of

pressing the wrong button and ruining their project. A trainer of several software programs commented about this.

“...and I think the search tools in all the software, especially, well not so much MAXqda but it may be, but in NVivo, N6 and ATLAS definitely people are scared to use the search tools because they're not the most intuitive things and that's a lot of what we do is try to go over what they all do and how it all works and sort of try to tell people you're not going to blow your computer up or ruin your project by asking questions ...”

It was often confirmed to Lewins in the help/observation sessions that at the root of much insecurity was the fear of experimentation and that intermediate or advanced support often became a session to reassure the user about how much was safe, and how few acts were 'dangerous' in software.

More serious amongst advanced users, or those further into the analytic process, were worries that they were not using the software effectively and to its full potential and that this was somehow impacting on the quality of work. For them the search tool and organisation of their coding schema were the main areas of insecurity. These issues were particularly noticeable amongst those who actively sought informal help and reassurance or took part in the observation sessions. Again reassurance was the key issue when dealing with concerns about coding schema; the researchers often created their coding schema in intuitive or methodologically aware ways, yet still felt under confident about manipulating codes or re-arranging ideas and collections of data for different analytic tasks. Often they just needed to be given a sense of 'secure freedom' to experiment along analytic paths they already knew they wished to take.

In questioning the value of her software based work one researcher undertaking a PhD study concerning the built environment was worried that using software tools in a confused way might lead her in the wrong direction-

“..if I've forgotten what I'm doing - what happens if I misuse the software and if I'm not sure I'm doing it right and it tells me that there's this wonderful link that everyone's going to go mad about, that link might not be there because I might have put it there by accident”.

There are two observations to make here about this extract. First, concerns about the analytic process itself were conflated with the insecurity about how to use software tools and second, these insecurities were exacerbated by something in the nature of an overdependence on the utility of the software, while she was outside her 'comfort zone'. Later, the same researcher is less explicitly concerned with software issues when she expresses some basic insecurity about qualitative data analysis itself and researcher bias –

“...I'm probably thinking long term but I've imposed these codes, how can I avoid the bias that I bring this project. Because these are my codes, how can I sort of verify?”

As we shall discuss later in the section on the effect of software on method, this user, acutely conscious of a particular problem, was more likely to produce robust analysis, than a researcher who was unaware about the nature and demands of qualitative data analysis itself.

In a more logistical sense, the historian, just starting out on her qualitative software career was to begin with very uncertain about how the coding process (and the manner in which she was doing it) was going to help later; yet as she grew more confident, her worries shifted to whether what she was achieving inside the software would be any use or accessible to anyone else should she not be available to explain things.

“..That's the other thing that worries me if I get run over by a bus, which is not impossible, that it's accessible to other people”

On different but related logistical issues, informal questions to the CAQDAS helpline were often about backing up files and also the transfer of projects from one computer to another. This aspect of file management was the single most common reported problem during the 3 month period of helpline logging (the help-line only saw the problems rather than the routine successes). Such problems were usually resolvable, but often quite complicated to fix especially for those users whose general computer awareness was low. Though these problems are purely practical in nature, the delays and anxieties they have caused have been considerable. On rare occasions projects became unworkable or weeks of work were lost because of unsystematic saving or inherent software problems in project transferral routines.

"... it went belly up for me just a wee while ago, and I lost about 2 or 3 days worth of materials and it was a very productive 36 hours of work, if that makes sense, so that was a shed load of stuff that I lost, so you can become a bit over reliant whereas if I'd have had post-it notes and my bits of paper and my A3 sheets then I'd have been able to pick up where I left off, short of a fire or a flood"

Isolation exacerbated nervousness about attempting new things or trying out fixes for sometimes quite small problems. The user below was without any immediate help close at hand to help her back up her project or move it -

"... like I said, I'm scared that I will screw it all up or something. I don't know how to zip; I have a zip thing on my computer but I don't even know how to do it in a Word program let alone take my N6 project and mess it up. And I know that I can take some things to Ann, but it's difficult, you know I'm here and she's there and we have to arrange a time and it's just not convenient you know. But she's about the only person that I know".

The notion of being overly-cautious about trying things out is understandable but it was a recurring and constraining theme which local support and/or more familiarisation either with the computer system or the software, would have fixed. In the above case a small nudge in the right direction would have fixed the problem and transformed her feelings of security.

ANALYSIS AND THE EFFECT OF SOFTWARE ON METHOD

The research team viewed this aspect of the data with special interest. The web resource being created needed to avoid appearing to make assumptions about users' methods or methodology; it needed to incorporate some sense of the relationship between methodology and software, and to do that it was useful to be aware of respondents' perceptions of how they were working, how specific they were about their methodological approach, and in relationship to that, whether the role of software was positive or problematic.

Much of the commentary was positive in respect of software flexibility and the enhanced management of data. Respondents commented on various aspects of data and idea management. A MAXqda user writes about her approach to managing literature: -

"Yes. Well talking again about the literature reviews that I do, the software.... I would say the software really supports the work that I do on those literature reviews because the approach that I take with those literature reviews is to summarise literature and research and so on as sort of entries as if I was entering them on to a data base, that's what I do, and then the software allows me to sort of split up those summaries into whatever codes and so on, variable that I want to look at, but then it still allows you to look at the whole as well".

A student comments on how ATLAS.ti helped to manage the connections between ideas:-

"... I think I found that quite good in my PhD stuff I used very much, sort of, grounded theory...way of using it and it really did help me kind of figure out what my ideas were about the [.....]. So I thought it was very good in, in allowing me to sort of develop theories and look at,

sort of, relationships between data and kind of build up... and that I didn't necessarily have a clear idea of even from having done all the interviews.."

An NVivo user referred to the process of adapting to the software but also the variable paths that could be taken with it

"...but I guess in some ways although the software doesn't learn adapt to you there has to be some kind of give or take or you just give up on the packagebecause NVivo is quite varied in the way you approach it, there's so many different ways of doing things, that's been really helpful"

and,

"In terms of it, it has been grounded theory it fits, but it's taken a wee bit of working to actually make it fit me, if that make sense?"

Several respondents were using both qualitative methods and software for the first time. One student, as was seen earlier, was rather conscious that the quality of her work might depend on her clear reading of the software tools and her correct use of them.

In other respects there were some reservations from several users, somewhat paradoxically connected with the way software makes it easier to work in certain ways.

"I think just generally using this kind of software it sometimes forces you to do things and it's a bit...it kind of forces you to sort of try and code things into sort of quite a linear way which sometimes seems to be a disadvantage"

Another researcher reports -

"I think also the biggest one is that you begin to construct data, or deconstruct data, using the software and that can be a limitation and I think you've got to recognise that...I think we get in to the habit of using the software as the way, if that makes sense, and everything fits in with it rather than actually pushing the boundaries that little bit further"

and similarly -

"...You become lazy. I think sometimes, sometimes you expect the software to do things, like I could always do a search and that'll will do or that will come up with enough".

Over-fragmentation of data has been seen as a questionable by-product of being able to code easily -

"...Sometimes it can...because it encourages you to chop things up and sort of see things in like little separate codes or whatever, you can sometimes forget to look back at the whole picture".

Again, in reporting similar perceptions from most of the observation sessions, the support Lewins often gave was concerned with encouraging the researcher to relax about over-fragmentation or large numbers of codes. The advice was often to see their value and retain them but also be ready and able to use search tools or 'collecting' devices to amalgamate them in varied ways in order to step back and study new selections or broad sweeps of data from different angles, and possibly compare across different subset information.

Another experienced user of one software package (NVivo) was forceful on the subject of developers advocating specific ways of coding and structuring coding schema to enhance efficiency.

“...but you know for a long time the people in the sort of computing side of QDA have been saying ‘now this doesn’t really change the way we do our analysis..... But now you’ve got people like (...) saying ‘No, no. The way we’re coding needs to change to use the software properly’. He’s saying ‘don’t code in this way code in this way’ and that’s...and now I’ve got warning bells starting to ring, you know, when a programmer, (.....) starts telling me that I need to have a coding structure thought out properly before I can get my analysis done so that I can use his search tool more effectively now I start to worry. Now I think well maybe the people warning us off computers, maybe they’ve got a point. But you know I think they’re both extreme positions”.

It is evident from the experiences of supporting users, from the observation sessions and seeing projects in progress that however researchers are recommended to structure coding schemas, they do work in an intuitive way to manage their ideas. What we found is that in general, often from a lack of time, software was used in quite simple ways and users’ management and contact with data was enhanced accordingly; developers and experts often have over-ambitious ideas how software should be used, and in the case of NVivo there has been a growing awareness that some of its key promoters recommend a particular, so called ‘ideal’ way of working with it, so that it becomes a normalized method. One of the disadvantages of this approach in such dominant software as NVivo is that such a method would not be methodologically neutral. In contrast, in developing the web resource what we wanted to facilitate was the notion that all QDA software could be valued more by researchers when regarded as tools which can be useful in many different ways to support differing methodological approaches.

It is also important to report, however, that some of the experienced researchers, especially in applied fields did not claim to be using any particular methodological design. There were at least four respondents who admitted consciously or unconsciously to a lack of methodological awareness. This user acknowledged that the software (MAXqda) provided a way to work:-

“..Well I don’t think my methodological approach... it wasn’t very sophisticated. It was...that was then lead by what the software could do, that kind of developed the methodology. The fact that you could put these codes on and do more sub codes so it was kind of using in that way really. Rather than me having ideas and then seeing if I can do it through the software.”

The ATLAS.ti user below revealed that for their purposes actually reading the data was a last resort. This may have been a circumstance dictated by the costs of coding in this particular applied situation, or a necessarily more surface level scrutiny of a very large dataset. It meant a dependence on a flawed quick-coding process in software, and a restricted idea of the qualitative model of work –

“Well the thing is, I mean what we found was, I mean although the auto coding is very good, you can you know just specify it, give it a word or two words and it would find all those words within, with all the document, that was excellent. But we found when we were having conversations or meetings etc people don’t tend to use, always use the same word, so it missed, some of them. So it wasn’t picking up all the quotations. That was one of the drawbacks and also, which also meant that we really needed to go back and read the document and code it up individually and that was time consuming”

Ideally this user wanted even more sophisticated tools to improve on the performance of the quick autocoding process. It is difficult to be sure whether her statement showed a lack of understanding about robust qualitative analysis, or whether what she actually needed was content analysis software to statistically analyse the content of her data.

CONCLUSION: OPERATIONALISING THE FINDINGS

In operationalising these findings to guide the design of the web resource we created, a variety of basic IT skill levels as well as the use of qualitative software were addressed. Step-by-step

processes at several very introductory stages were designed to supplement those who were self-reliant from forced circumstances or from their own volition. For later use, more advanced processes were covered in a more discursive way to encourage lateral thinking and flexible use of software to increase the value and manipulation of users' coding schema; for example, to encourage flexible use of existing coding schema. Logistical support aimed to encourage systematic saving processes, backing up and safe project transferral. We also developed suggested protocols to enable forward planning for teams planning to work with software in more challenging ways.

From the perspective of the Online QDA and CAQDAS projects, it may be understandable that the clear implication of the qualitative element of the user needs aspect of the project was particularly welcome, in that it suggested there was a need for a continually available source of support with an institutional identity formed around availability without the need for special provision or special pleading by researchers using the software.