Learning From The Learners.

The Role Of Technology Acceptance And Adoption Theories In Understanding Researchers' Early Experiences With CAQDAS Packages

Christina Silver, Christine Rivers

Abstract

This paper presents findings of a qualitative longitudinal study tracking the use of different CAQDAS tools over the period of 12 months. This is the first project of its kind that follows researchers from learning a CAQDAS software to applying skills and using it in a research project. Findings illustrate that initial enthusiasm with the potential of software is often tempered by frustrations with its actual use. Users frequently attribute frustrations and cessation of use to lack of software functionality. However, successful adoption of CAQDAS technology is related to methodological awareness, adeptness in the techniques of analysis and technological understanding. Theories of technology acceptance and adoption have been used to contextualise findings and to develop a CAQDAS-specific model that helps teaching CAQDAS software.

Acknowledgments

We would like to thank participants for sharing their experiences and time that helped to collect this type of data. Their contribution is invaluable to the CAQDAS community.

This work was supported by the Economic and Social Research Council (ESRC), National Centre for Research Methods (NCRM): [RES-576-25-0002]. This study formed part of research undertaken by the Qualitative Innovations in CAQDAS (QUIC) between 2008 and 2011. QUIC formed part of the UK-based CAQDAS Networking Project (CNP) which has provided information, advice, training and on-going support in the use of CAQDAS packages since 1994. Key aims of QUIC were to develop accessible online materials that support researchers in learning and using CAQDAS packages, and to improve all levels of training provision. Results outlined in this paper contributed to the development of the extensive resource materials available from the 'Outputs and Guidance' part of the CNP Website. Silver designed and led this research but both authors contributed equally to the work underlying this research article.

Keywords

Qualitative Data Analysis, CAQDAS, Technology Acceptance, Technology Adoption, User Experience.

Introduction

Qualitative data analysis (QDA) is a multifaceted process (Mason, 2002) requiring knowledge about the philosophy and theory of qualitative research and the specific techniques of analysis (Bazeley & Jackson, 2013; Richards, 2002; Maxwell, 1992). It does not demand software although informed use thereof can assist efficient and effective processes. Appropriate and effective use of software requires methodological knowledge (Richards & Richards, 1994) and analytical skills (Kuckartz, 2001; Richards, 2002; Johnston, 2006, Davidson & Jacobs, 2008) but there is often a lack of access to local expertise in QDA and software (Fielding & Lee, 2002; Lewins & Silver, 2007). The teaching of qualitative research methods is in itself complex (Breuer and Schreier 2007); methodology informed teaching of software adds a further layer of intricacy (Crowley, Harre & Tagg, 2002).

This study illustrates the interrelated issues involved in how researchers learn about and use qualitative software. Focusing on the use of CAQDAS packages during the year following initial training, we report how participants evaluate software training and incorporate aspects of instruction into working practices. Use of tools at different stages of work and the challenges experienced are discussed. Learning about and using software are shown to be inherently related to knowledge about the theory of qualitative research ('methodological awareness'), adeptness in the techniques involved in conducting analysis ('analytic techniques') and confidence with the use of technology generally ('technological savviness').

Effective use of CAQDAS involves translating the needs of a given study (informed by methodology and analytic techniques) into the practices of using software tools to achieve the research aims (Woolf, 2013). Students often learn qualitative theory, methodology, practical techniques and software tools concurrently (Gibbs, Friese & Mangabeira, 2002). Many confusions and complaints concerning software tools revealed by this study relate to frustration with the interplay between these phases. Software is sometimes criticized for lacking capabilities which in fact are present. Rather than being attributable to technological functionality, many either do not know how to undertake aspects of analysis in line with their methodological and theoretical framework, or are insufficiently conversant with the software to do so.

Teaching And Learning CAQDAS

There is increasing literature concerning the use of CAQDAS technology (White & Polandri 2012; Gibbs, 2013). Reflexive pieces provide insights into how researchers' learn about and use particular functionalities from an 'insider' point of view. Some accounts reflect on the appropriateness of different products for specific analytic strategies (King, 2010; Carcary, 2011; Kus 2011). Others discuss the role of software in QDA more generically (Mavrikis & Geraniou 2010; Odena 2012). In contrast to largely positive evaluations, some have criticised the suitability of CAQDAS for certain approaches. MacMillan (2005) for example, in comparing NVivo, MAXQDA and Qualrus, claims they added little value to the outcomes of her discourse analysis; suggesting they cannot help with the organization of materials required for in-depth and in-context analysis. This standpoint reflects her perception of the interplay between the techniques required by her analytic strategy and the tools provided by software. That she was unable to use software tools effectively for her approach, however, results from a failure to manipulate them appropriately rather than the unsuitability of these packages for discourse analysis per se. Indeed, these approaches are well supported by CAQDAS (Silver & Fielding, 2008; Ryan, 2009).

Trainers, methodologists and developers tend to write instructive pieces that emphasize the importance of planning for the use of software and its role in ensuring high quality analysis. For example Lewins & Silver (2007) and Di Gregorio & Davidson (2008) focus on efficient and effective use of software in qualitative research referring to different levels of usage, expected usefulness and expected ease of use. They comment on common analytic and technical challenges likely to influence usage and in the longer-term,

acceptance. There has, however, been little research into trends of CAQDAS usage since Fielding & Lee (2002), although Hughes, N.G., Lewins, A. & Silver, C. (2010) , White & Polandri (2012), Fielding, N., Fielding, J. & Hughes, N. G. (2012), and Gibbs (2013) include some data. The way users learn about CAQDAS is an even more neglected area which this study addresses.

Theories Of Technology Acceptance And Adoption

The acceptance and adoption of technology (Dix, Finlay, Abowd & Beale, 1998) has been studied from a range of perspectives (Dillon & Morris, 1996) and many models to account for and predict initial and sustained use have been developed and tested (Chuttur, 2009). Davis' (1993) technology acceptance model (TAM), see Figure 1, is widely cited and used as the basis for predicting system use understanding motivations.

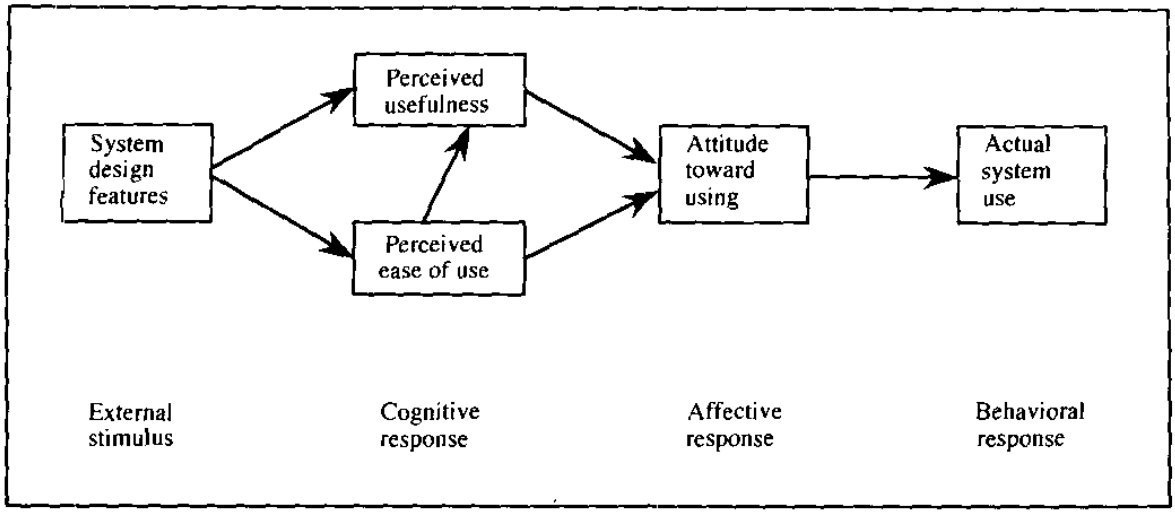

Figure 1: Davis Technology Acceptance Model (1993:476).

Perceived ease of use ("the degree to which a user believes that using the system will be free from effort") and perceived usefulness ("the degree to which a user believes that using a system will enhance his/her performance") are identified as cognitive responses influencing attitudes towards IT systems and their actual use. System design features are identified as external stimulus impacting upon both perceived usefulness and perceived ease of use. The original theory suggests a relatively straightforward causal relationship between ease of use and perceived usefulness, with both being directly influenced by system characteristics.

Davis' model has been frequently discussed, replicated and empirically verified (Adams, Nelson & Todd, 1992; Hendrickson, Massey & Cronan 1993), critiqued and adapted (Davis 1993; Venkatesh & Morris, 2000; Burton-Jones & Hubona 2006; Bagozzi 2007), mainly through the use of quantitative methods. Amongst its modifications are the inclusion of behavioural intention to use as a variable directly influ-

enced by perceived usefulness (Davis, Bagozzi & Warshaw 1989; Davis & Venkatesh 1996); the inclusion of antecedent variables to perceived usefulness and perceived ease of use (Vankatesh & Davis 2000; Venkatesh 2000); and discussion of the social aspects involved in technology acceptance (Venkatesh & Morris, 2000; Børn, Fitzgerald & Scopula, 2003). However, few qualitative studies have used or tested the model (notable exceptions being Neville & Fitzgerald 2002; Børn, Fitzgerald & Scopula, 2003; Renaud & van Biljon 2008).

Factors involved are shown to vary according to the technology and population of users under consideration and more complex models than Davis' 1993 version have been developed in specific contexts. One such example which has particular resonance to our study is shown in Figure 2. In their Senior Technology Acceptance & Adoption Model (STAM), Renauld & van Biljon reflect the nature of the population and technology they studied: 'elderly mobile phone users'. In particular they take account of how these users initially appropriate mobile phone technology (fewer than half their sample bought their mobile phones themselves) and account for the fact that not all users eventually accept and adopt the technology (rarely acknowledged in other models).

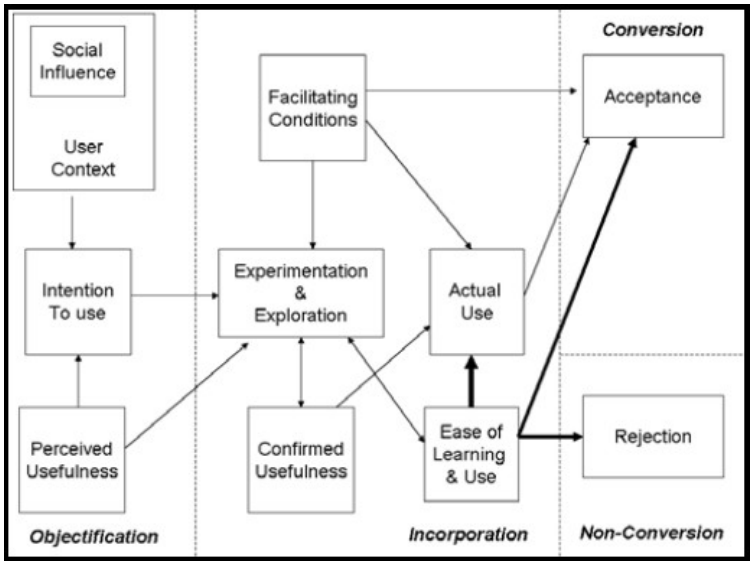

Figure 2: Senior Technology Acceptance & Adoption Model (STAM), Renauld & van Biljon 2008.

Such studies, theories and models provide insights into why certain types of users adopt and accept certain types of technology, but they do not reveal how they learn to use it. In addition, acceptance is illustrated as an outcome, whereas, as discussed in this paper, in the context of CAQDAS a certain level of acceptance of the technology is required in order to engage with tools. Although elements contained within technology acceptance models have been alluded to by some authors in discussing CAQDAS (Carcary, 2011; Friese, 2011; Woods & Dempster, 2011; Schoenfelder, 2011; MacMillan, 2005), its use has not previously been discussed specifically in these terms. In using TAM and derivatives to contextualise findings we illustrate important insights into how students learn about CAQDAS packages, highlight limitations of existing theories of technology acceptance and adoption in understanding their experiences, and present our own model that reflects the contexts relevant to CAQDAS technology.

Method

The main body of this study comprised 23 participants. Each attended a 2-day introductory software workshop in one of five packages of their choice. Respondents received three online questionnaires over the 12 months following training. Table 1. shows, for each software workshop, the respondents participating in each of the three waves of questionnaires, and indicates the rate of attrition.

	Wave 1 (1 month after training)	Wave 2 (6 months after training)	Wave 3 (12 months after training)
ATLAS.ti	6	5	3
MAXQDA	5	2	0
NVivo	4	4	3
QDA Miner	4	3	2
Transana	4	2	2
Total	23	16	10

Table 1: Sample and attrition

Several factors may have contributed to the rate of attrition, including project termination, change in project researchers and software cessation. Each participant worked on an existing project, used a specified analytic approach[1] and one of the five CAQDAS packages. This paper considers common experiential themes present regardless of the software product or analytic approach employed. Due to the small number of participants appropriate for this qualitative study, we do not speculate on the prevalence of the various themes between software packages. Most respondents were doctoral students and we therefore focus on this population.

Questionnaires comprised closed questions about use of specific tools designed to make direct comparisons of software usage over time. Open-ended questions sought more detailed comments on experiences and evaluations. The latter are the main subject of discussion here. After completing all three questionnaires participants were offered a project-specific session to provide on-going support and to gain further understanding of their progress with software. Two respondents took up this opportunity. Their sessions were recorded, transcribed and integrated with the questionnaire analysis. These data are augmented by our observations at software training events over a total of more than 15 years.

Data were analysed using a software-supported thematic approach, comprising data familiarization, coding, reflection, theme generation and relation of themes to literature (Braun & Clarke 2006; Silver & Lewins, in press).

1Specified approaches were Grounded Theory, Conversation Analysis, Interpretative Phenomenology, Narrative Analysis, Critical Discourse Analysis, Content Analysis, Ethnographic Approaches and Nomothetic & Thematic Analysis. Some had already used other packages or previous versions before attending training, thus the sample comprised different levels of prior CAQDAS knowledge.

Findings And Analysis

Learning CAQDAS: Evaluations Of Initial Training And Impact On Practice

The model of two-day introductory training reported on here is designed to facilitate researchers in three key areas: i) understanding the place of software in the broader context of qualitative research; ii) gaining a broad and comprehensive overview of the product, illustrating potentials for data management and analysis across a range of analytic approaches; iii) enabling participants to design a strategy for effective use in the context of their own project needs.

Experiences Of Software Training: Formats, Expertise And Interactions

Wave 1 questioned respondents' experiences and evaluations of training; including levels of satisfaction with particular aspects; whether training had or was likely to change practice; what were considered the most and least useful aspects; whether documentation had been of use; and suggestions for improvement. Training was widely evaluated positively and enthusiastically; the majority reporting it to have been 'very useful' for setting-up software projects and getting started with analysis[2]. Most frequently mentioned in elaborating the usefulness of training in open-ended responses were 'hands-on practice', 'practical demonstration' and the 'expertise of trainers'.

Facilitating participants in translating learning into practice is a key aim of training and responses illustrate the benefits of this approach. Respondents expressed the value of this model clearly, often in terms of planning for future use and applying learning effectively:

> Being able to work on our own data on the second day and discuss this with the tutors was incredibly useful and allowed us to think through how we were going to use [the software] with the data we have and how [it] can be used throughout the project…

> The combination of teaching and individual coaching was ideal. It meant that I was able to check my own understanding in the context of my own research so I left the course feeling 100% confident that I knew either a) what I needed to do or b) who I could ask if I got stuck.

The perceived expertise of the tutors was mentioned frequently, appearing to be closely related to participants' levels of satisfaction with the training and sense of confidence with the software. Trainer expertise was seen as valuable in focusing attention on particularly useful software tools, on identifying shortcut ways of locating functions or executing analytic tasks and in providing a comprehensive overview of functionality.

The benefits of face-to-face training were also mentioned in this context. This speaks to the importance of learning in 'traditional' ways for many, illustrating that despite increasing availability of online software

2Both these questions were asked using a Likert-type scale where 1 = very useful and 5 = not at all useful. In the responses to both these questions 3 was the lowest score recorded.

training opportunities, face-to-face models retain an important role. Indeed, several respondents were reflective about their own learning styles, explicitly commenting on aspects of training in terms of how it enabled them to learn effectively. For some this was related to the interaction between trainer and participant. For others it was related to learning alongside others, in which interaction with other participants played an important role in their overall experience of the training, their thinking about their own work and in their retention of information.

Perceived Usefulness Of CAQDAS Packages: Impact Of Training On Intentions

Closely related to respondents' satisfaction with training is their general perception of the usefulness of software. This was captured by responses to several questions, including one which asked whether any aspect of respondents' practice was likely to change as a result of the training. All bar two responded positively to this question, often providing specific examples of how they planned to use software. Of these responses the majority included specific reference to data management and organizational issues, with several indicating this to be a key aspect taken away from the training. For example:

> I will be far more organised going into it and have actually used my pilot data manually to help me organise it prior to putting it in to the programme. Not something I would normally do as I tend to jump in with software and work it out as I go. The importance of being organised came across so strongly that I wanted to get a handle on some initial coding from the pilot to work out what should be codes, what should be families and how I could / would link them etc. By doing this manually to start it enabled me to get my file naming etc in a sensible state prior to uploading documents.

A significant amount of pre-preparation was undertaken in order to ensure for an effective plan of analysis, illustrating this respondent had understood the importance of early preparation on future potentials. Several also commented that training had increased their awareness about the potential of software and the extent to which it would be useful to them in the longer term. This was the case both for those who had little or no prior experience with software and those who had used it to some extent already. As such training had a clear impact on consolidating intentions to use CAQDAS.

Comments such as the following indicate the effectiveness of training in illustrating general software potential and in helping participants decide how to use it for their own projects:

> I didn't realise the full capabilities of [software] and wasn't really sure how to use it for my project but think that I do now.

This illustrates both a generic acceptance of CAQDAS technology prior to training attendance, and that training content consolidates this attitude. Some indicated an expectation that software would enhance their analytic practice in some way. For the respondent who made the following comment this was related to ensuring analytic processes are rigorous. She expected this may be easier to achieve when using software than working manually:

> Having used cards and highlighters in the past, I feel using a software program will make the process more thorough and it is also excellent for recording my progress etc.

The role of training is not contained within Davis' original TAM or its derivatives, but is a key aspect of consideration in the context of CAQDAS technology. Renaud & van Biljon (2008) account for 'experimentation and exploration' which impacts on 'confirmed usefulness' and 'ease of learning and use', but in the context of mobile phone adoption formal training of the technology does not take place. Many attend training having already begun collecting data and are keen to get started formally with 'analysis'[3]. Others attend speculatively, to establish whether a given package provides tools suitable for their needs. Yet others attend at an early stage of their research, sometimes with the intention of incorporating software into project planning. Whatever the situation of individual researchers, at least some level of acceptance of the potential value of CAQDAS technology is implied by training attendance. Participating in training which encourages the use of participants' own research materials and focuses on enabling participants to design strategies for effective software use grounded in their own research design also implies a broad pre-existing intention to use the technology. Findings from wave 1 illustrate that training improves participants' knowledge about functions, perceived usefulness and thereby consolidates intentions to use. Perceived usefulness (evidenced in enthusiasm about software and confidence it its potential) was a common theme throughout wave 1 and was closely related to the perceived effectiveness of the introductory training. However, this confidence did not always continue once independent use of software became more established.

Using CAQDAS: From Perceived Usefulness Via Ease Of Use To Actual Use

CAQDAS packages offer tools that facilitate QDA. Coding tools are fundamental but are augmented by others designed to support the processes of integration, organization, interrogation, visualization and reflection (Lewins & Silver 2007; Silver & Lewins, in prss). In investigating how researchers use software it is crucial to understand how they move beyond early tasks of project set-up to employ more sophisticated tools in the context of their own project needs. This study shows that enthusiasm and confidence subsequent to introductory training may not always be maintained. This was often conceptualised by respondents as resulting from inadequacies of software, but is actually related to the fact that most students attending these workshops were learning about the theory and techniques of QDA simultaneously with software tools. Contrasting perceived usefulness of software during guided learning via training and during the early stages with actual independent use of software over time is of particular interest in understanding both the utility of tools and in targeting training.

3Not within the scope of the discussion here, but important to note, is the issue of discrete stages or phases of analysis and the role of software in integrating them (see Silver & Lewins, forthcoming).

Getting Started With Software And Main Uses In The Early Stages

Wave 1 included questions designed to uncover how respondents transfer knowledge gained at training into practice as well as how they generally experienced early stages of independent work with software. Responses indicate use of specific tools and expressions of their value, according to how soon after training respondents had begun using software independently and how frequently they were using it.

First is the issue of 'immediacy of independent use'. More than half reported having started using software within 1-3 days of training. 4 respondents had not yet started using software at the time of Wave 1, usually because they had not yet collected data, with the remaining 4 having started between 4 days and 4 weeks after training. That most started using software soon after training indicates they were eager to use their chosen product, that they had research materials ready to work with and that training had inspired them to get started quickly. However, four respondents commented that they had not yet used the software 'enough'. Detailed reasons for this are not known, but such comments suggest they believe transferring knowledge gained at training into practice is likely to be more effective the 'sooner' and 'more' they work.

Although most who took part in this study began using software independently soon after training, this is not always the case. Indeed, it is not uncommon for participants to attend introductory training for a second time, or for those attending intermediate/advanced training to have used software only to a limited extent. Training sessions provide a large amount of information during a relatively short period of time, and consolidation of knowledge may be particularly challenging for those who attend as complete novices – either in terms of software exposure or qualitative analytic expertise.

In terms of software exposure, the few respondents who had some experience with software prior to training identified this as being beneficial to their learning experience. For example:

> I thought the balance and range of topics covered was really good. I know I benefited as I'd played with [the software] before I went to the course so I was familiar with it and got a huge amount out of the day; it may be worth considering making the online […] tutorials mandatory before attendance so that students don't waste time familiarising themselves with the terminology when they could be seeing it used in the context of research.

The issue of participants' 'readiness' for software training is complex, and in large mixed ability and background groups which are typical of these sessions, it is difficult, if not impossible, to ensure participants start from the same point.

The challenges experienced by novice analysts in consolidating knowledge gained at training into effective independent use was clearly illustrated by one respondent who took advantage of the support session after completing the wave 3 questionnaire. In describing her project at the outset of the session she stated that:

> ...the project is a qualitative study [...] and its obviously quite individual really and based around in depth interviewing to try and really capture what something is like really and to some degree in

> terms of support but in each case they sort of went into the history a bit of what happened in terms of diagnosis and things, and that was quite useful to find out in relation to what their expectations of support were too I think. So I wasn't absolutely sure at the beginning what approach I was going to take to the analysis really [...] but I've now decided because, because of what I've heard really that a narrative approach seems a, they seem to be telling a story, so looking at the narrative approach seems a good idea.

Not only was she unable to describe her research design concisely, but that she had settled upon an analytic approach (that she could only broadly specify) after 12 months of software use indicates a lack of understanding of the interplay between analytic techniques (as informed by methodology) and software tools. She had worked with the software quite extensively but her work lacked direction. Her early use of software was motivated by enthusiasm at its potential use, as derived from the training, but her lack of QDA expertise meant she struggled to employ the software effectively once she was doing so independently.

Questionnaires asked a series of closed questions regarding the use of specific tools. Of those common to all packages the most frequently used at this early stage were code and retrieve tools and those relating to the organisation of respondents and data to known characteristics, such as socio-demographic attributes. Given the characteristics of respondents' projects in terms of analytic approach and the extent to which training had been well evaluated this is not surprising. Most respondents' stated analytic approaches were code-based and therefore early experimentation with and focused use of the range of coding tools is to be expected. Indeed, qualitative comments from Wave 1 frequently mentioned coding tools in terms of their perceived usefulness and ways they were actually used. Most expressed a high level of satisfaction with coding tools and their long-term potential, although a few were daunted by the importance of early coding choices on later work or felt constrained by the particular coding functionality of their chosen package (see below).

The early use of data organisation tools evidenced in Wave 1 reflects the emphasis of their importance in training, indicating the value of creating an organisational framework early-on resonated with respondents. Indeed, several referred to software as a project management tool; a key focus of training. Respondents' working practices seem to have been directly influenced by this way of conceptualising qualitative software. One respondent explicitly stated that training consolidated his intentions to use:

> Had not used [the software] before this, so training will of course change practice! More practically, I'm more likely to use more advanced analysis tools beyond basic coding following training.

Several explicitly mentioned using software to help manage their literature review, plan the structure of the final report or thesis and make linkages between literature and data[4]. For those with prior experience with manual QDA, potentials for organising ideas and data through the use of software were prioritized.

4The logic and use of CAQDAS packages in this sense is described fully in Silver & Lewins 2014, which forms the basis of the structure of introductory software training at the CAQDAS Networking Project.

That the training was still fresh in their minds is evident in Wave 1 responses, with several expressing enthusiasm concerning the prospect of applying what had been learnt to their own research projects.

Changing Perceptions Of Usefulness

Although most were very positive about the potential of software in wave 1, responses became more detailed and critical in later waves. With increased length of use comes increased familiarity with functionality and criticism where weaknesses are identified. In particular, data collected 6 and 12 months post-training reveal frustration where respondents perceive functionality to be 'missing'.

Confusions, Inadequacies And Criticisms

Increasing confusion concerning tools and criticisms about functionality initially seems surprising as it may be assumed that increased familiarity with software would increase confidence with functionality and reduce frustration at having to experiment to 'get it right'. Most had begun using software soon after training, so it is unlikely that the general experience of increasing confusion and criticism is solely related to 'immediacy of use'.

Increased use of software was sometimes coupled with uncertainty about the functioning of tools and their potential:

> I have found all of these 'potentially' useful. So for example in my basic output files when a memo is linked to a quote it is useful that it can appear to remind me. However, I feel that I am missing something and that I have not understood the full benefit of these tools

This comment was made after working with software for 12 months. She was doing a PhD and thus developing her analytic skills in tandem with learning to use software and this was proving difficult at times. So much so that she subsequently carried out some aspects of analysis using a spread sheet application rather than persisting with the CAQDAS package. She subsequently took advantage of the support session at which she outlined her confusions in detail. During the session the trainer clarified functionality and provided ideas for work-around solutions to her difficulties. As a result she went back to using the software. Some learners may not be able to investigate the intricacies of software functionality independently; needing external confirmation that they are 'doing it right' and suggestions for next analytic steps. This respondent's experience is not atypical and clearly indicates some of the issues associated with learning about and using qualitative software effectively, particularly for novice analysts. She displayed knowledge about what she wanted to achieve generally, couched in terms of her research questions, but lacked awareness of or confidence in the techniques to do so. Her independent experimentations with software tools were therefore unsuccessful and she blamed this on perceived inadequacy of the software. It was evident during the support session, however, that she was unclear about how to achieve her research aims, and this was a function of her general lack of awareness of and expertise in the techniques

of analysis. Her use of software uncovered this. However, that she took up the support session indicates a desire to continue use and reveals acceptance of its potential usefulness.

Several other respondents also expressed a lack of confidence in their abilities and/or a lack of knowledge of the software, helping explain the decreased sense of usefulness of software as time progressed. As illustrated in the following quote, taken from Wave 3, this was sometimes expressed in relation to the way earlier tasks had been undertaken:

> I feel as if I don't know half of its usefulness and do not have the confidence that I have set up the initial coding to enable me to make to best of the tools.

A lack of confidence is expressed here in relation to software tools but this comment indicates a lack of clarity around the techniques of analysis. The result is difficulty in understanding the potential value of software tools, and hence inability to use them efficiently. Lack of confidence, frequently expressed in terms of software tools (for example implications of coding functionality, query tools and output options) often reflected general absence of methodological awareness and/or specific absence of clarity or adeptness in the procedures of analysis. For example, some reported difficulty linking codes to express relationships and, crucially, understanding how this would be analytically useful.

One respondent was particularly concerned about getting started with coding when she realized the extent to which later work would be based on earlier coding:

> I did not dare go into coding when I realised how reliant I would have been on the tool itself and did not feel competent enough to risk it as I did not have access to IT support at work and would have been badly limited if things went wrong.

Such unease illustrates that using software independently can be daunting. For this respondent a lack of general IT confidence was closely related to her feelings about using a CAQDAS package and the complexity of software was a source of concern. Her perceived lack of on-going support compounded her unease. In addition, she could not find a way of using coding tools to support her analytic approach and as a result abandoned use:

> … using a Grounded Theory approach with line-by-line coding seems prohibitive for using the software as I found it too onerous to try and use the coding and went instead back to doing it by hand…

That most packages do not enable working with fixed lines as contextual units is a common frustration, especially amongst those used to working manually. However, it indicates a basic lack of understanding of how coding works in software and that the purpose of numbered lines of textual data as a referencing device is largely redundant in the software environment. This respondent reported being experienced in conducting QDA without software but struggled in her attempts to replicate manual analytic methods within software. She had initially been enthusiastic about the potential of software use but her perception of the ease of use changed considerably over time, ultimately leading to cessation of use. Lack of

confidence in technology generally, or 'technological savviness', is an issue in itself in learning to use CAQDAS tools. Coupled with lack of clarity about the methodological underpinnings of analytic tasks or their analytic manifestation compounds the issue; of which this respondent's experience is a powerful example. However, she valued the potential of the tool sufficiently to train a research assistant in its use:

> As I had a research assistant (student) during the summer I was able to show her how it works from the hand-outs so she was trained up in its use ... for inputting the hand coding I have done.

Her experience illustrates the complex processes involved in learning about and using software; her strength of belief in the potential of software (her broad acceptance of its worth) was sufficient to find a workable solution to continued use despite frustrations with specific features.

As time passes requirements of software move from basic project, data and idea management to the execution of particular analytic tasks. Those that have a more precise conceptualisation of their analytic needs, and the end to which they are being put, appear to struggle less with working out how to use software effectively. For the students in this sample that were learning about QDA concurrently with learning the practicalities of software use, confusions and frustrations were compounded. Experimentation is required to appropriately manipulate software tools to conduct specific analyses and to query data in depth. Where there is a lack of clarity around the techniques involved in a specific type of QDA such experimentation is less likely to be fruitful. As such, frustration ensues.

Common Analytic Challenges

As well as general confusions concerning software functionality respondents raised a number of analytic challenges encountered in using software; the most common of which pertain to two of the key aspects involved in code-based QDA. We broadly categorize them here as 'working efficiently with coding schema structures' and 'moving on from coding tasks to more analytic stages of work'.

Working With Coding Schema Structures

Developing efficient coding schemes is inherent to QDA of the kind supported by CAQDAS. Those working inductively tend to initially generate large numbers of detailed codes, needing to refine the coding scheme later to make sense of how it represents data and relates to theory. Generating excessive numbers of codes is discussed elsewhere (Coffey et al, 1996; Richards, 2005; Lewins & Silver 2007; Rivers & Bullock, 2011) with some suggesting that software encourages the generation of 'too many' codes. Issues related to coding inductively using software was clearly expressed by one respondent who had previous experience of coding manually:

> I am not sure [the software] has enabled me to think in an unexpected way but it has made me more 'wordy' and I think my initial coding has been more thorough, possibly a bit excessive compared to the way I coded with cards prior to using the software...

That he sees software as facilitating a more *thorough* coding process and encouraging the development of *more* codes than would otherwise have been the case is interesting. The ease with which to revisit coded data, un-code and re-code when using software facilitates the development of categories and of checking their validity. When coding manually the generation of large numbers of very detailed codes may be avoided due to the relative difficulty in revisiting and comparing coded data.

Those working deductively may seem to have more control over coding schemes, at least in the early stages, although they usually want to allow for identifying themes not included in the *a proiri* framework (Lewins & Silver, 2007) and therefore also encounter the need to restructure. Data gathered via the questionnaires, interviews and from observing researchers attending intermediate/advanced training indicate that issues related to generating satisfactory coding schemes are encountered whatever the analytic approach. For some issues are related to the physical structure of schemes as they appear in software, for others to the linking of codes and the expression of relationships. Others have more practical concerns. The following quote again illustrates a lack of clarity around analytic tasks; suggesting this respondent explored software tools, trying to work out how they might be useful, rather than having an analytic need clear in her mind and then looking for software tools to satisfy it:

> [I] have attempted to generate [a…] view of coding structure. Currently a bit unsure about why to generate relationships amongst the codes and what they mean.

Such issues were mentioned most extensively in Wave 2, when respondents had been working with software for about 6 months. This is to be expected as during this time-frame it is common to be deeply engaged with coding work. These complaints are more refined than those discussed above, indicating a more in-depth experimentation with software tools. However, they illustrate the challenges involved in learning the techniques of QDA concurrently with the software tools that can be manipulated to achieve them.

Amongst critiques of CAQDAS is the idea that code-based tools encourage thematic data fragmentation at the expense of other means of organizing data and ideas about them (Coffey at al. 1996). Such concerns are not infrequently encountered during training sessions, and were articulated in this study:

> ... I have become worried about becoming too reductionist and loosing the effect of 'wholeism'. I am dealing with stories and the coding has allowed me to identify the themes etc but sometimes I feel I have lost the complete message of the story (unless I created a code for the whole script I suppose).

Those adopting narrative-type approaches often struggle with this aspect of software use. However, as seems to be the case in this example, difficulties usually relate to how software is conceived and used, rather than the functionality of particular tools or the *emphasis* of a given product. Indeed, this example illustrates the connection between intended application of analytic strategy, software tools, their perceived usefulness, and actual use. Coding tools can be used in various ways to navigate around and access data. They may be employed in traditional, methodologically informed ways. Conversely, those less

wedded to particular approaches may choose to be flexible and creative with regards to their use. Terminology is relevant here. Methodologically, the term 'coding' in qualitative analysis may have a particular meaning. Technically, and practically, in terms of software usage, this need not be the case. Various tools are provided by CAQDAS packages, but the way they are used is determined by individual users or teams. This is emphasised and illustrated in training, but comments such as these, illustrate that this is not always fully understood amongst participants.

From Coding To Analysis

Users struggle with moving on from initial coding to working more analytically to varying degrees. Wave 2 and 3 questionnaires were designed to elicit the issues and several participants' responses indicate this aspect of software use to have been problematic. Researchers need to be confident enough in their analytic strategy to creatively manipulate software tools to facilitate analytic work. The amount of time likely to have lapsed between training, where analytic tools are demonstrated, discussed and practiced, and actual need for them by users working independently, can be too lengthy for some. This appears particularly true for the methodologically inexperienced who may be unsure about the role of coding in analysis, and those that are technically reticent; reluctant to experiment with software tools for risk of making unrecoverable errors.[5]

CAQDAS packages provide a variety of analytic tools. Here we discuss query tools as an example of the type of issues encountered. Going beyond basic code and retrieve functionality requires an understanding of the technicalities of software functionality and the value of its role in analysis. Possessing both enables users to utilize tools appropriately and effectively. Progressing beyond coding can be experienced as challenging due to a lack of knowledge:

> I have tried a few basic queries but have not gained any huge benefit from these - again I feel it is user error / lack of knowledge.

Implicit in having experimented with query tools is an awareness of their potential, and this respondent is reflective about her software knowledge. It might be that at the point of experimenting with query tools she had done insufficient coding to illustrate their utility, or indeed, that she was using them incorrectly. Either way, she persisted with experimentation because she felt "*there is a lot of potential that I am exploring".* This respondent was relatively new to QDA and her comments imply she was to an extent looking to the software to show her 'potential'. As discussed above in relation to another respondent, the simultaneous learning about the methods and the tools causes issues.

In contrast, some respondents provided specific examples of analysis they would like to conduct but had not found a way of achieving within the software, for example:

5 Indeed, intermediate/advanced workshops provide project-specific support in these aspects, and were attended by some of the sample reported on here.

> I would really like to know how to/if it's possible to sort the coding by date in some fashion, so I can see how press coverage/people in the press coverage refer to specific issues as the story of [….] progresses, and broader long term trends in the data. Similarly I have ended up (so far) doing my quantitative coding (which newspaper, page numbers, type of journalist, section, attitudes […]) in an Excel chart, as at present it is a bit hard to see how to integrate the two types of analysis.

> The next steps in making the most of the coding I have already completed as I feel it has the potential to tell me more about what I have found out such as developing patterns between participants responses and identifying differences in the way codes are expressed.

Both the integration of qualitative and quantitative analyses and the identification of patterns across responses are enabled by CAQDAS packages, but statements such as these indicate the subtlety of the relationship between analytic need and software functionality. Both these respondents had clear ideas about what they needed to achieve but were struggling with finding ways within software to do so. The issues they raise are commonly encountered when moving beyond initial coding to working at more analytic levels. Both these respondents were students. They subsequently succeeded in using software for these tasks, but their experiences indicate the ebb and flow involved in successful software use. As one respondent put it: "*doing & learning at the same time is slow work".* It is common for software users to be learning the tool at the same time as doing a real analysis. For students the issues are compounded because they are typically concurrently learning about qualitative methodologies more broadly, and their philosophical underpinnings. The practical application of the theory in terms of the actual tasks that need to be undertaken can in themselves be difficult to uncover. Doing so within a complex CAQDAS package which offers a multitude of pathways adds further complication and therefore confusions, frustrations and feelings of inadequacy are understandable. These issues are also evident in comments relating to the need for on-going support in the use of software.

Discussion: The Interplay Between Technology Acceptance And Processes Of Adoption

This study illustrates a number of tensions in how CAQDAS packages are learned about and used; relating to perceptions of their potential, the actual experience of their use and (not always accurate) knowledge about the existence of and subtleties within particular tools. Existing theories and models are useful in contextualising the processes involved but are

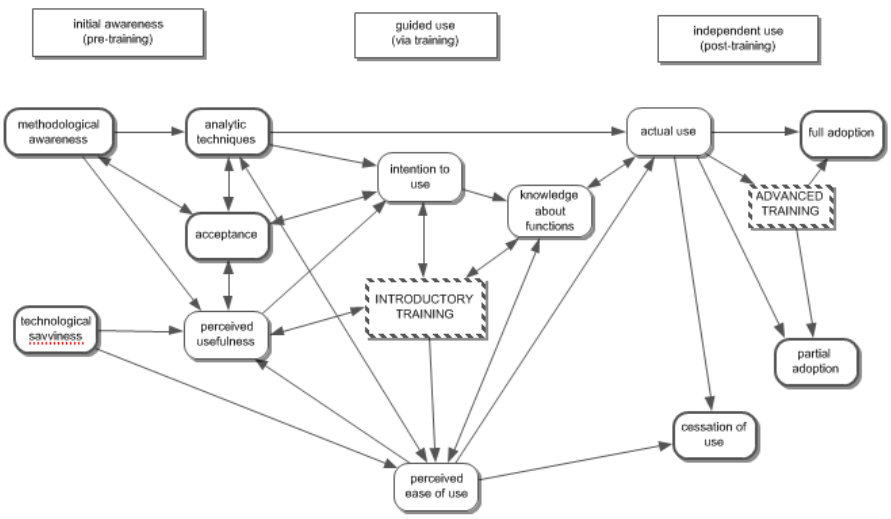

Figure 3: Technology acceptance and adoption amongst CAQDAS users

insufficient in explaining the use of CAQDAS technology for two principle reasons: i) the purpose of CAQDAS; and ii) the interdependent factors involved in successful use. CAQDAS packages are designed to support researchers in the analysis of varied types of projects and as such they do not necessarily promote any particular approach to analysis. The range of tools provided by individual packages are not all required for any given analysis, and therefore users must decide which tools to use for particular analytic tasks, and how those procedures relate to the broader project needs. Successful use is related to methodological awareness and analytic techniques, both of which exist independent of software, as well as perceived usefulness and ease of use of the technology itself.

We therefore propose a CAQDAS-specific model (Figure 3) which borrows elements of others but takes account of the nature and purpose of qualitative software, the role of training and the intentions and expertise of users. Specifically, i) the processes of technology acceptance and adoption operate cyclically as opposed to causally; ii) perceived usefulness and a level of acceptance is a pre-cursor to training attendance and is affected by the content and nature of instruction; iii) the nature of independent use (whether full or partial adoption, or indeed cessation of use) is affected by the interplay between methodological awareness and analytic requirement and levels of technological savviness; iv) software use may be effective despite adoption not being full; and v) full adoption may require multiple trainings.

Perceived usefulness, ease of use and actual use are elements of TAM (Davis 1993). 'Methodological awareness', 'analytic techniques' and 'technical savviness' are *external factors* particularly relevant to the way users engage with software initially and which mediate perceived usefulness and ease of use. Technology acceptance is an attitude (Renauld & Biljon 2008), which mediates and is influenced by all other aspects in the model. Adoption of CAQDAS technology is a process from 'initial awareness' (which usually occurs before training) to 'full adoption'. Adoption is not conditional upon receiving training, although where it is attended (as was the case with the sample of respondents discussed in this paper), it consolidates perceived usefulness and influences intentions to use. Training effectiveness, perceived ease of use and actual use are influenced by general awareness of the philosophy of qualitative data analysis, adeptness in the specific techniques involved in conducting analysis and technical savviness as well as knowledge of software tools. Training does not always result in adoption. Training may be undertaken multiple times before successful adoption is achieved. However, CAQDAS technology need not be fully adopted to be used effectively – i.e. not all the tools in a given software need to be used for an analysis to be conducted effectively.

As Figure 3 illustrates, this study suggests the relationship between ease of use, perceived usefulness and actual use of CAQDAS packages operates circularly rather than in a logical causal direction, as originally proposed by Davis' (1993). CAQDAS packages are by no means fully entrenched in qualitative practice. 'Perceived usefulness' is both a precursor to attending training and consolidated as an outcome of it. However, acceptance of CAQDAS technology does not necessarily result in its (successful, efficient or otherwise) *adoption.* Initial enthusiasm can be tempered by the experience of actual use. As time pro-

gresses some experience more difficulty in software use and confusion about and misunderstanding of functionality. This is particularly evident amongst students learning about QDA in tandem with software tools. Issues are often couched in terms of an inadequacy of software, but indicate a lack of understanding of or confidence with the procedures of analysis. This highlights the complexity of the process of learning, accepting and adopting CAQDAS technology. As discussed by Renaud & Van Biljon (2008), acceptance is an attitude towards technology, influenced by other factors.

External and personal factors are important dynamics in understanding how some overcome these issues whereas others do not. With CAQDAS technology, confidence with technology generally (technological savviness), the ability to carry out analysis using an existing method (the dynamic between methodological awareness and analytic technique), as well as software knowledge (gained through training and actual use) all play a role. Methodological awareness is possessed in varying degrees prior to attending training, affecting motivation to attend, perceived usefulness and actual use. The higher prior methodological awareness and understanding of analytic need, the easier participants are likely to find translating software tools to the needs of particular projects. Technological savviness is observed as affecting initial engagement with software, comprehension of computational logic, readiness to experiment and anxiety relating to perceived effective use. Lack of prior experience has an impact on levels of technological savviness.

Methodological awareness, adeptness in analytic techniques and technological savviness operate on continuums. In terms of technological savviness this is likely relatively straightforward, in that the more experience and confidence learners have with technology generally, the more likely they are to feel comfortable with the technical aspects of learning a CAQDAS package. With respect to methodological awareness and adeptness in analytic techniques, however, the dynamic is more complicated. Some may have very little awareness of the theory or strategies of QDA at all, and essentially learn about the practicalities of undertaking analysis through the medium of the software. Others may be very experienced in a particular methodological approach, but are undertaking a project using a different analytic strategy and struggle to work in alternative ways. Yet others may be very experienced in a particular methodological approach and have clearly defined analytic needs, but find software difficult to adopt for those requirements. Those with a high degree of technological savviness, methodological awareness and adeptness in analytic techniques - in an approach to which CAQDAS are well suited - may be likely to both *accept* CAQDAS technology and *adopt* it more readily.

Conclusion

CAQDAS technology has been commercially available since the late 1980s. Its use is not universally entrenched in qualitative analytic practice although, as evidenced by increasing discussion of use acceptance and adoption appear to be increasing and, as such, its utility more widely acknowledged. However, the tensions revealed by this study are important for developers and teachers to consider. Although many

go on to use software effectively, for some the frustrations experienced act as significant barriers, with the extreme result potentially being cessation of use. The range and complexity of tools increases as software develops. This can result in novice researchers and software users finding pathways to robust analysis difficult to achieve. The interplay between general methodological awareness and adeptness in specific analytic techniques are therefore crucial and needs to be fundamental to how CAQDAS is developed, taught and discussed.

References

Adams, D. A., Nelson, R. R., & Todd, P. A. (1992). Perceived usefulness, ease of use, and usage of information technology: a replication. *MIS Quarterly*, 227-247.

Bagozzi, R. P. (2007). The Legacy of the Technology Acceptance Model and a Proposal for a Paradigm Shift. *Journal of the Association for Information Systems*, 8(4), 3.

Bazeley, P., & Jackson, K. (Eds.). (2013). *Qualitative Data Analysis with NVivo*. Sage Publications Limited.

Bjørn, P., Fitzgerald, B., & Scopula, A. (2003). The role of social awareness in technology acceptance of groupware in virtual learning teams. In *Proceedings of the 26th Information Systems Research Seminar in Scandinavia* (IRIS, Haikko Manor, Finland).

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, *3*(2), 77-101.

Breuer, F. & Schreier, M. (2007). Issues in Learning about and Teaching Qualitative Research Method and Methodology in the Social Sciences. Forum: Qualitative Sozialforschung, Forum: Qualitative Social Research (8)1 http://www.qualitative-research.net/index.php/fqs/article/view/216 (Date of access: 18.5.2011 1:19 pm)

Burton-Jones, A., & Hubona, G. S. (2006). The mediation of external variables in the technology acceptance model. *Information & Management*, *43*(6), 706-717.

Carcary, M. (2011). Evidence Analysis using CAQDAS: Insights from a qualitative researcher. The Electronic Journal of Business Research Methods (9) 1. pp.10-24

Carvajal, D. (2002). The Artisan's Tools. Critical Issues When Teaching and Learning CAQDAS [47 paragraphs]. *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, *3*(2), Art. 14, http://nbn-resolving.de/urn:nbn:de:0114-fqs0202147

Chuttur, M. (2009). Overview of the technology acceptance model: Origins, developments and future directions.

Coffey, A., Holbrook B. & Atkinson P.(1996) 'Qualitative Data Analysis: Technologies and Representations. Sociological Research Online (1)1.

Crowley, C., Harré, R., & Tagg, C. (2002). Qualitative research and computing: methodological issues and practices in using QSR NVivo and NUD* IST. *International Journal of Social Research Methodology*, *5*(3), 193-197.

Davidson, J., & Jacobs, C. (2008). The Implications of Qualitative-Research Software for Doctoral Work Considering the Individual and Institutional Context. *Qualitative Research Journal*, *8*(2), 73-80.

Davis, F.D. (1993). User Acceptance of Information Technology: System Characteristics, user Perceptions and Behavioural Impacts. *Int. Journal Man-machine Studies*. 38. pp. 475-487.

Davis, F. D., Bagozzi, R. P., & Warshaw, P. R. (1989). User acceptance of computer technology: a comparison of two theoretical models. *Management Science*, *35*(8), 982-1003.

Davis, F. D., & Venkatesh, V. (1996). A critical assessment of potential measurement biases in the technology acceptance model: three experiments. *International Journal of Human-Computer Studies*, *45*(1), 19-45.

Di Gregorio, S. & Davidson, J, (2008). Qualitative research design for software users. Open University Press, Maidenhead.

Dillon, A. & Morris, M. (1996). User acceptance of information technology: theories and models. Annual Review of Information Science and Technology. p 3-32. Medford NJ: Information Today, Inc.

Dix, A., Finlay, J., Abowd , G. & Beale, R. (1998). Human-Computer Interaction, second edition. Prentice Hall.

Fielding, N. & Lee, R. (2002). New patterns in the adoption and use of qualitative software. *Field Methods*, 14 (2), pp. 197-216.

Fielding, N., Fielding, J. & Hughes, G. (2012). Opening Up Open-Ended Survey Data Using Qualitative Software *Quality and Quantity* May, Online First Articles.

Friese, S. (2011). ATLAS.ti for Analysing the Financial Crisis Data. Forum: Qualitative Sozialforschung, Forum: Qualitative Social Research (12)1 http://www.qualitative-research.net/index.php/fqs/article/viewArticle/1632 (Date of access: 25.5.2011).

Gibbs, G. R., Friese, S., & Mangabeira, W. C. (2002, May). The use of new technology in qualitative research. Introduction to Issue 3 (2) of FQS. In *Forum Qualitative Sozialforschung/Forum: Qualitative Social Research* (Vol. 3, No. 2).

Gibbs, G (2013) 'Using software in qualitative analysis' in Flick, U. (Ed.), 2014. The Sage Handbook of Qualitative Data Analysis. Sage Pubns Ltd.

Hendrickson, A. R., Massey, P.D., & Cronan, T. P. (1993). On the Test-Retest Reliability of Perceived Usefulness and Perceived Ease of Use Scales in *MIS Quarterly* , 17(2). pp. 227-230.

Hughes, N.G., Lewins, A. & Silver, C. (2010) Document per Case vs Document per Question: Tuesday 4 January 2011A discussion on whether to structure open-ended survey data by creating a 'document per case or a document per question' for analysis in CAQDAS packages. http://www.surrey.ac.uk/sociology/research/researchcentres/caqdas/support/analysingsurvey/document_per_case_vs_document_per_question.htm (Date of access: 28.07.11)

Johnston, L. (2006). Software and method: Reflections on teaching and using QSR NVivo in doctoral research. *International Journal of Social Research Methodology*, *9*(5), 379-391.

King, A. (2010): 'Membership matters': applying Membership Categorisation Analysis (MCA) to qualitative data using Computer-Assisted Qualitative Data Analysis (CAQDAS) Software, *International Journal of Social Research Methodology*, 13:1, 1-16.

Kuckartz, U. (2001). An introduction to the computer analysis of qualitative data. Sage.

Kuş S. E. (2011). Systematic Versus Interpretive Analysis with Two CAQDAS Packages: NVivo and MAXQDA [75 paragraphs]. Forum Qualitative Sozialforschung / Forum: Qualitative Social Research, 12(1), Art. 34, http://nbn-resolving.de/urn:nbn:de:0114-fqs1101345

Lee, R. & Fielding, N. (1996). Qualitative Data Analysis: Representations of a Technology: A Comment on Coffey, Holbrook and Atkinson' *Sociological Research Online*, (1) 4 http://www.socresonline.org.uk/1/4/lf.html (Date of access: 18.5.2011).

Lewins A., & Silver C. (2007). Using Software in Qualitative Research: A Step-by-Step Guide. Sage Publications. London.

Mavrikis, M. & Geraniou, E. (2011): Using Qualitative Data Analysis Software to analyse students' computer-mediated interactions: the case of MiGen and Transana, *International Journal of Social Research Methodology*, 14:3, 245-252.

MacMillan, K. (2005) More than just Coding? Evaluating CAQDAS in a Discourse Analysis of News Texts. Forum: Qualitative Sozialforschung, Forum: Qualitative Social Research (6) 3. http://www.qualitative-research.net/index.php/fqs/article/view/28 (Date of access: 24.5.2011).

Mason, J. (2002). *Qualitative researching*. Sage.

Maxwell, J. A. (1992). Understanding and validity in qualitative research. *Harvard educational review*, *62*(3), 279-301.

Neville, K., & Fitzgerald, B. (2002). An Innovative Training Model for an Organization Embracing Technology. *Journal of Information Technology Education: Research*, *1*(1), 193-200.

Odena, O. (2012). Using software to tell a trustworthy, convincing and useful story. *International Journal of Social Research Methodology*, (ahead-of-print), 1-18.

Renaud, K., & Van Biljon, J. (2008, October). Predicting technology acceptance and adoption by the elderly: a qualitative study. In *Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries: riding the wave of technology* (pp. 210-219).

Richards, L. (2002). Qualitative computing--a methods revolution?. *International Journal of Social Research Methodology*, *5*(3), 263-276.

Richards, L. (2005) Handling Qualitative Data: A Practical Guide. Sage Publications. London.

Richards, T. J., & Richards, L. (1994). Using computers in qualitative research. *METHODS*, *1*, 1.

Rivers, C. & Bulloch S. (2011). CAQDAS A contributor to social scientific knowledge. In: Methods Spring NEWS 2011. p.2.

Ryan, M. (2009) 'Making visible the coding process: Using qualitative data software in a post-structural study', Issues in Educational Research, 19(2): 142-161.

Silver, C. (2010) Research design for longitudinal case-study project : tracking the use of software in real projects using different methodologies. QUIC Briefing Paper, Qualitative Innovations in CAQDAS (QUIC), The CAQDAS Networking Project, University of Surrey, UK. http://www.surrey.ac.uk/sociology/research/researchcentres/caqdas/files/BriefingPaperResearchdesignforlongitudinalprojectFINAL.pdf (Date of access: 10.06.11)

Silver, C. & Patashnick, J. (2011). Finding Fidelity: Advancing Audiovisual Analysis Using Software [88 paragraphs]. Forum Qualitative Sozialforschung / Forum: Qualitative Social Research, 12(1), Art. 37, http://nbn-resolving.de/urn:nbn:de:0114-fqs1101372. (Date of access: 10.06.11)

Silver, C. & Lewins, A., (in press) Computer assisted analysis of qualitative research

Schoenfelder, W. (2011). CAQDAS and Qualitative Syllogism Logic – Nvivo 8 and MAXQDA 10 compared. Forum: Qualitative Sozialforschung, Forum: Qualitative Social Research (12) 1. http://www.qualitative-research.net/index.php/fqs/article/view/1514 (Date of access: 18.5.2011).

Venkatesh, V., & Davis, F. D. (2000). A theoretical extension of the technology acceptance model: four longitudinal field studies. *Management Science*, 46(2), 186-204.

Venkatesh, V. & Morris, G. M. (2000). Why don't men ever stop to ask for directions? Gender, Social Influence, and their Role in Technology Acceptance and Usage. *MIS Quarterly.* 24(1). 115-139.

Woods, D. K., & Dempster, P. G. (2011, January). Tales from the bleeding edge: The qualitative analysis of complex video data using Transana. In *Forum Qualitative Sozialforschung/Forum: Qualitative Social Research* (Vol. 12, No. 1).

White, M.J., Judd, M.D., Poliandri, S., 2012. Illumination with a Dim Bulb? What Do Social Scientists Learn by Employing Qualitative Data Analysis Software in the Service of Multimethod Designs? *Sociological Methodology*, 42, 43–76.

Christina Silver

Dr. Christina Silver is manager of the CAQDAS Networking Project (CNP). The CNP provides information, advice and training in a range of software packages designed to facilitate the analysis of qualitative data. She leads the training and capacity building activities of the project and had a lead role in research conducted under the National Centre for Research Methods (NCRM) funded project Qualitative Innovations in CAQDAS (QUIC). Christina's particular interests relate to the relationship between technology and methodology and the integration and analysis of non-textual data, using software. She is also co-director of the University of Surrey's Day Courses in Social Research programme and co-founder of Qualitative Data Analysis Services (QDAS), which provides customised consultancy services for individuals and groups engaged in qualitative and mixed-methods analysis.

Christine Rivers

Dr Christine Rivers is a Teaching Fellow in the Surrey Business School and Associate Lecturer in the Department of Sociology at the University of Surrey, UK. The author graduated from the University of Vienna, Austria in 2005 with a Masters in Media and Communication and earned her PhD in Human Computer Interaction from the University of Surrey, Faculty of Engineering in 2010. Christine's research includes exploring innovations in teaching and learning and the use of technology as teaching and learning device as well as the investigation of smartphones as a marketing tool.

Article Information

This article is published at the Digital Repository of Technische Universität Berlin, URN urn:nbn:de:kobv:83-opus4-44300, http://nbn-resolving.de/urn:nbn:de:kobv:83-opus4-44300. It is part of ATLAS.ti User Conference 2013 : Fostering Dialog on Qualitative Methods, edited by Susanne Friese and Thomas Ringmayr. Berlin: Universitätsverlag der TU Berlin, 2014, ISBN 978-3-7983-2692-7 (composite publication), URN urn:nbn:de:kobv:83-opus4-51577, http://nbn-resolving.de/urn:nbn:de:kobv:83-opus4-51577