

Contents

<i>List of Figures</i>	xii
<i>List of Tables</i>	xv
<i>List of Boxes</i>	xvii
<i>Companion Website</i>	xxi
<i>Acknowledgements</i>	xxii
<i>About the Authors</i>	xxiii
Introduction	1
Some personal history	1
Our thinking	3
Why a second edition?	4
Chapter overview	5
Chapter exercises and the companion website	7
Our ultimate aim	8
Chapter 1: Qualitative Data Analysis and CAQDAS	9
Qualitative research and data analysis	10
The practicalities of research in the software context	12
Managing and referencing literature	13
Formulating the research problem and defining the research questions	13
Representing theoretical frameworks	14
Incorporating research materials	14
Defining factual features	14
Developing analytical areas of interest	15
Some basic principles and distinctions	16
Analytic <i>processes</i>	16
Levels and directions of work	17
Code-based and non-code-based approaches	18
Cuts through data	19
The rise of qualitative software	20
What types of software do we categorise as CAQDAS?	21
Which is the 'best' CAQDAS package?	22
Analytic strategies in the context of software use	22
Analysis of discourse	24
Narrative inquiry	25
Framework analysis	26
Grounded theory	27
Thematic analysis	29

Mixed methods research	30
Visual analysis	32
Concluding remarks: a critical yet flexible approach	33
Chapter 2: The Nature of Software Support for Research Projects	35
The project management potential of CAQDAS packages	35
Starting points	36
Familiarisation	36
The software project as a container for your work	37
Case-study examples	37
Case study A: Young People's Perceptions	38
Case study B: The Financial Downturn	43
Case study C: Coca-Cola Commercials	44
Qualitative activities and software tools	44
Integration of sources and analyses	46
Exploration of content and structure	48
Organising materials and ideas	51
Grouping	51
Coding	52
Hyperlinking	53
Reflecting upon data, interpretations, processes and results	53
Retrieve, review and rethink data and ideas about them	54
Memo, summarise, track, output	55
Connecting and visualising interpretations	55
Interrogating to identify, compare and test	56
Identifying patterns, relationships and anomalies	56
Comparing subsets and cases	57
Testing theories and assessing quality	58
The right tools for the job	59
Concluding remarks: flexibility in the sequencing of tasks	59
The bits in between	59
Chapter 3: Software Summaries	61
ATLAS.ti	61
Dedoose	64
HyperRESEARCH	66
MAXQDA	68
NVivo	70
QDA Miner	72
Transana	74
Resources	76
Chapter 4: Data and their Preparation for CAQDAS Packages	79
Data types	79
File formats	81
Textual formats	82

Multimedia formats	83
Quantitative formats	83
Textual data preparation	84
Data structures	84
Units of recognisable context	86
Transcription guidelines for textual data	88
Are special formatting considerations really necessary?	92
Structural coding without auto-coding (no special formatting)	94
Formal transcription conventions	95
Multimedia data preparation	96
Social media	97
Direct or indirect handling	97
Assistance for transcribing – and developing synchronised transcripts	99
Mixed data	100
Descriptive or quantitative data import	101
Pre-coding – survey data import and auto-processing	102
Concluding remarks: laying the groundwork	104
Exercises: data and their preparation	104
Chapter 5: Early Steps in Software: Practical Tasks and Familiarisation	106
The way work can happen	107
Gain familiarity with software by setting up a project	107
Creating the project	107
Transparency	108
Naming and backing-up routines	109
Incorporating research materials	111
Getting the software project and the interface shipshape	113
Project design	115
Early organisational structures for data	115
The virtue of empty places for thinking and growing	119
Creating a framework of memos	121
The first memo	122
The dispersal of notes around the project	123
Ideas for naming memos effectively	123
Overt reflections and reflexivity: thinking out loud; telling the story	124
Memos attached to other entities	125
Standalone memos – as project management devices	126
Scoping the topic area and critiquing the literature	126
Exports from customised literature management tools	127
Optimising tools for literature management	129
Concluding remarks: groundwork for efficient analysis	131
Exercises: getting started	131
Chapter 6: Exploration and Data-level Work	134
Early exploration of data	135
Familiarisation during early handling	136

Marking data for relevance and significance	137
Simple data reduction devices and workarounds	138
Annotation tools – their universal utility	139
Multimedia data: annotations and data reduction	142
Annotating data – to aid continuity, reflexivity and openness	145
Quick content searching tools	146
Word frequency tools in CAQDAS packages	147
Text or lexical searching – the practicalities	148
Text-mining tools and complex pattern searching	150
Hyperlinking	152
Practical aspects of hyperlinking	154
Concluding remarks: appropriate use of data-level tools	155
Exercises: exploration and data-level work	156
 Chapter 7: Qualitative Coding in Software: Principles and Processes	 158
What is qualitative coding?	158
How coding works in qualitative software	158
Approaches to coding	160
Induction, deduction, abduction: logics of reaching explanations	160
Coding terminology	161
Inductive approaches to coding	162
Deductive approaches to coding	166
Theoretical coding	168
Question-based coding	168
Combining approaches: the practice of abductive coding	
strategies using software	170
The flexibility of combining approaches	174
Coding visual data: 'indirect' and 'direct' approaches	175
Coding visual data 'indirectly' via synchronised transcripts	175
Coding visual data 'directly', without an associated transcript	177
Coding in software, whatever the approach	180
Bases for generating codes	181
Concluding remarks: using software to support <i>your</i> approach to coding	182
Chapter exercises	182
 Chapter 8: Basic Retrieval of Coded Data	 186
Principles of basic retrieval	187
Purposes of basic retrieval	188
Aiding continuity: where did I get to last time?	188
Aiding continuity: generating snapshots of coding status	188
Moving the analysis on: identifying areas for further consideration	189
Moving the analysis on: recoding	191
Moving the analysis on: comparing coding	192
Types of basic retrieval	193
Quantitative overviews	193
Horizontal cuts	195

Vertical cuts	198
Simple filtering devices for early comparative interrogations	200
Generating output	200
Concluding remarks: reflexivity and rigour	202
Chapter exercises	203
Chapter 9: Working with Coding Schemes	205
Breaking down data, building them back together	206
Structures of coding schemes in software	207
Functioning and implications of hierarchy	208
The behaviour of hierarchical coding schemes	208
Non-hierarchical systems	210
What type of coding scheme will suit the way you work?	210
Creating coding schemes	211
Project-related factors influencing the development of coding schemes	212
Escaping the confines of coding scheme structures	214
Separating areas of the coding scheme for pragmatic or theoretical reasons	214
The relationship between the coding scheme and the theoretical framework	215
Better ways to express and collate theory – mapping and short-cut groupings	219
Coding scheme maintenance – routine actions	221
All codes, all data	223
Concluding remarks: manipulating coding schemes for your needs	225
Exercises: managing and manipulating coding schema structures	226
Chapter 10: Managing Processes and Interpretations by Writing	229
The importance of writing in analysis	230
Writing as a continuous analytic process	230
Forms, purposes and spaces for writing	231
Appraisals	231
Field notes	232
Transcriptions	233
Annotations	233
Definitions	235
Analytic memos	236
Process memos	238
Summaries	240
Final write-ups	240
Considerations when writing in software	241
Managing your writing	244
Creating, naming and dating	244
Grouping memos	246
Structuring writing	248

Integrating your writing with the rest of your work	248
Linking writing	249
Visualising memos	249
Coding your own writing	251
Searching the content of your notes	251
Outputting writing	252
Concluding remarks: integrating writing	253
Exercises: managing processes and interpretations	254
Chapter 11: Mapping Ideas and Linking Concepts	257
Mapping traditions and other software	258
Other types of 'mapping'	259
Purposes of mapping in CAQDAS packages	259
Mapping to express theoretical connections	261
General mapping functionality in CAQDAS packages	262
Software-specific functions and specialities	264
Remembered vs. scribbled links	264
Working at the data level within maps (ATLAS.ti and MAXQDA)	266
Creating, hiding and revealing layers in maps (MAXQDA and NVivo)	267
Visualising co-occurring codes in maps (ATLAS.ti and MAXQDA)	270
Creating codes (and other project items) in a map (ATLAS.ti and MAXQDA)	271
'Intelligent' links and functional relationships in maps	273
Concluding remarks: extensive possibilities for mapping	275
Exercises: mapping ideas and linking concepts	276
Chapter 12: Organising Data by Known Characteristics	278
The importance of good organisation in reflecting project design	279
The earliest basics of organisation – and the limits	279
Timing: when to put more complex organisational structures in place	282
Illustrating the potential at the interrogation stage	283
What does a data file consist of?	284
Circumstances, conditions, contexts, cases	286
The evolution of data organisation	286
Imperfect categories	288
Case studies	289
Organising whole documents in software	290
Organising at document level – step-by-step advice	292
Organising at document level – by importing a spreadsheet (or survey)	293
Starting a table off in the right format?	294
Organising within the document (parts of documents)	294
Coding in step-by-step ways	297

Auto-coding	297
The implications of coding cases, respondents and parts of files in terms of their further organisation	298
Concluding remarks: potentials and cautions	299
Exercises: organising data by known characteristics	300
Chapter 13: Interrogating the Dataset	303
The role of interrogation in moving on	303
The incremental, iterative and repeatable nature of querying	304
Combining different dimensions of data	305
Test theories and expectations (hunches)	306
Creating signposts for and from queries	307
Identify patterns and relationships	309
Compare subsets, cases and interpretations	311
Quality control	314
Quality: queries improving interpretive processes	314
Quality: flag up problems and check work	315
Software tools for interrogating the database	315
Searching content and/or structure	316
Simple forms of retrieval	317
Readily available information about codes (without building complex queries)	319
Coding queries	320
Qualitative cross-tabulations	322
Visualising results	323
Tables and matrices	323
Charts and graphs	326
Concluding remarks: interrogation functionality in CAQDAS packages	327
Chapter exercises	328
Chapter 14: Convergence, Closeness, Choice	333
Planning for the use of software	333
Convergence of tasks and tools: software as a container for your work	334
Closeness to data: inside software and outside it	335
Changing techniques of data analysis	335
Automation, quantitisation and mixing methods	336
Visual and social media analysis	337
Focused effective use of software	338
<i>References</i>	339
<i>Index</i>	346