

README FIRST FOR A USER'S GUIDE TO

Qualitative Methods

Third Edition

*To our students, with thanks for all they have
taught us about the challenges of learning to work qualitatively.*

Lyn Richards

RMIT University

Janice M. Morse

University of Utah

 **SAGE**

Los Angeles | London | New Delhi
Singapore | Washington DC

FOR INFORMATION:

SAGE Publications, Inc.
2455 Teller Road
Thousand Oaks, California 91320
E-mail: order@sagepub.com

SAGE Publications Ltd.
1 Oliver's Yard
55 City Road
London EC1Y 1SP
United Kingdom

SAGE Publications India Pvt. Ltd.
B 1/1 Mohan Cooperative Industrial Area
Mathura Road, New Delhi 110 044
India

SAGE Publications Asia-Pacific Pte. Ltd.
3 Church Street
#10-04 Samsung Hub
Singapore 049483

Copyright © 2013 by SAGE Publications, Inc.

All rights reserved. No part of this book may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from the publisher.

Printed in the United States of America

Library of Congress Cataloging-in-Publication Data

Richards, Lyn.

Readme first for a user's guide to qualitative methods /
Lyn Richards, Janice M. Morse. — 3rd ed.

p. cm.
Includes bibliographical references and index.

ISBN 978-1-4129-9806-2 (pbk.)

1. Social sciences—Research—Methodology.
2. Qualitative research. I. Morse, Janice M. II. Title.
- III. Title: Read me first for a user's guide to qualitative methods.

H62.M6612 2013
001.4'2—dc23 2012001310

This book is printed on acid-free paper.



Acquisitions Editor: Vicki Knight
Associate Editor: Lauren Habib
Editorial Assistant: Kalle Koscielak
Production Editor: Laureen Gleason
Copy Editor: Megan Granger
Typesetter: C&M Digital (P) Ltd.
Proofreader: Kate Macomber Stern
Indexer: Wendy Aliex
Cover Designer: Garon Kiesel
Marketing Manager: Nicole Elliott
Permissions Editor: Adele Hutchinson

12 13 14 15 16 10 9 8 7 6 5 4 3 2 1

Brief Contents

Preface	xvii
About the Authors	xix
1. <i>Why Readme First?</i>	1
PART I. THINKING RESEARCH	21
2. The Integrity of Qualitative Research	23
3. Choosing a Method	49
4. Qualitative Research Design	87
PART II. INSIDE ANALYSIS	117
5. Making Data	119
6. Coding	149
7. Abstracting	169
8. From Method to Analysis: Revisiting Methodological Congruence	185
PART III. GETTING IT RIGHT	213
9. On Getting It Right and Knowing if It's Wrong	215
10. Writing It Up	231
PART IV. BEGINNING YOUR PROJECT	253
11. Groundwork for Beginning Your Project	255
12. Getting Started	269
Appendix 1. Qualitative Software: Where to Go Next <i>by Lyn Richards</i>	281

Note from Tables 4.2 and 4.3 that qualitative programs differ greatly in the ways they can link with quantitative programs.

Think of this choice process as another step in the pursuit of methodological congruence. Just as research purposes and questions fit with data types and analysis strategies, so do software tools fit, for better or worse, with all these aspects of research design. Start there, setting out what you are asking, what data you expect to be handling, and by what methods of analysis, and then ask which of the tools available in software would best assist you.

The task is less daunting now than it was in earlier stages of software development. When computer tools were first designed for qualitative research, very different types could be identified (Tesch, 1990; Weitzman & Miles, 1995). Two decades later, there is a substantial common ground for basic functions, summarized in the regularly updated comparisons at the CAQDAS Networking Project website (<http://www.surrey.ac.uk/sociology/research/researchcentres/caqdas/index.htm>).

The main commonalities and differences at the time of writing are summarized in Tables 4.1 through 4.4.

<i>Expect This of All Qualitative Software</i>	<i>Look for These Differences</i>	<i>When Will This Matter?</i>
Provision for storage and managing of data and interpretations in a single unit or project	Programs differ in how a project is saved, stored, and transported.	Researchers nervous about security, ease of backup, and sharing of team projects should check for software help.
Ways of combining and comparing projects	Most programs support merging of projects, but they differ in flexibility of merging parts of projects.	This will matter if you have multiple researchers or sites or a good reason for combining your own projects.
Ways of backing up and safely storing projects	Programs differ in whether the source data is imported to the software or remains external to the software.	Mode of storage, if not understood, can imperil a project—be very clear about what should be backed up!
Ways of interfacing with other software	Programs differ greatly in whether they are designed to import from and export to statistical or database software.	This will be critical if you plan mixed methods research.

Table 4.2 Your Data Documents, Ideas, and Links

<i>Expect This of All Qualitative Software</i>	<i>Look for These Differences</i>	<i>When Will This Matter?</i>
Handling of text data for the project	Programs differ in permitted file formats (plain text, rich text, or specialized formats, including pictures, tables, etc.).	Text formatting matters most for projects with "rich" records.
Ability to create and edit text within the project; ability to type memos	Programs differ in whether data documents can be freely edited once "in" and in the flexibility of editing memos.	Typing up in the project matters most for records you want to code or annotate as you create them.
Inclusion of text data files in the project	Programs differ in whether they import the documents or link to files kept externally.	Where the data are stored may matter for security and convenience.
Handling nontext data—photos, videos, etc. (either importing and coding directly or ways of representing nontext records)	Some but not all will import nontext (pictures, video, or audio). Others vary in terms of what can be done with multimedia data.	This is a central concern if your design requires detailed analysis of nontext data.
Storing information (such as demographics) about people or places, etc.	Most will import such information from spreadsheets or statistics software but differ in options and display and in the flexibility with which you can use this information.	This is critical if you are doing mixed methods research or have a large sample.
Creation and editing of documents and memos from within the program	Programs differ in flexibility to edit and in ways memos are created and whether and how they are searched. Check whether the program is designed for extensive editing or just for corrections.	This is important if your method requires constant reflective records as theories are built (e.g., grounded theory).
Annotating or commenting on text	Programs differ in ways of annotating particular passages of your data, how annotations can be viewed, and how they are reported.	This is important if your method requires fine-detail commenting on discourse in texts (e.g., discourse analysis).
Support for linking to data within and outside of project	Programs have very different approaches and methods of linking: Look for what can be linked and how.	This matters if your method requires you to bring data together in ways other than coding.

Table 4.3 Coding and Text Search

<i>Expect This of All Qualitative Software</i>	<i>Look for These Differences</i>	<i>When Will This Matter?</i>
Coding of selected data into categories created by the researcher (called "codes" or "nodes") and retrieving all data coded into a category	Programs differ in mode of selection of data and procedure of coding. Some programs allow the researcher to record weighting of coding.	Coding style and facility are important to most researchers—try out software to see if you like the way it codes!
Ability to view all the data you have coded at a category	Programs differ in how you view coded data and whether and how the context can be retrieved. They also differ in how you can work with coded data to revise coding and optionally code further from it.	This matters most for methods where coding is just a first step toward interpretation, especially if it is important to explore the dimensions of a category.
Ability to see on the screen what coding you've done (usually in margin, sometimes by highlighting or reporting)	Programs differ in whether all coding can be seen at once and how the markings can be used to explore that code.	It will matter if you rely on (and are concerned about) coding or if in teams you want to compare coding.
Autocoding of data (mechanical finding and coding of words or segments)	Programs differ in how easily this is done and how much formatting is required, as well as whether you can set the context you want coded.	This is particularly important to projects with a lot of very structured data or requiring immediate retrieval (e.g., everything said by a particular speaker).
Text search of words in data and sometimes coding of the findings	Programs differ in the ways they conduct searches and store results, and whether you can save searches and results.	This is important if your method requires the mechanical processes of word search and/or further exploration of results.
Counting of codes or occurrences of words; quantitative content analysis	Some offer word frequency counts and quantitative reporting of searches, and some the ability to create your own dictionary.	This is important if your method requires counts. If so, check out "text retriever" and "content analysis" programs.

Table 4.4 Abstracting, Modeling, Questioning, and Reporting

<i>Expect This of All Qualitative Software</i>	<i>Look for These Differences</i>	<i>When Will This Matter?</i>
Management and viewing of coding categories	Most programs support hierarchical cataloguing of categories for review and access.	This is important if your coding categories will be numerous and/or if you are sharing coding schema.
Asking questions (with "search" or "query" tools) about patterns in the coding of data	Programs differ in ability to search combinations (e.g., Boolean, proximity) of coding, text, and the characteristics associated with people or places. Some allow multiple searches producing matrices for pattern exploration.	Some methods require sophisticated searches (e.g., matrices to show patterns). Do a "walkthrough" of the project to check how you want to query your data.
Saving of search results	Some save them as reports only, while others allow the possibility for the search results to be incorporated in the database.	This matters if you want to build inquiries on your coding patterns.
Ability to run repetitive searches	Only some programs provide for the researcher to write scripts to set up analysis processes.	This is important for projects where computer searching must be adapted to the design.
Visual displays	Most provide some tool for modeling. These vary greatly from simple diagrams to live data representations of theories and networks.	If visual representation of what is happening in your project is important, check what you need the software to display.
Ways of seeing connections you have recorded in the data	Some packages have tools to get you "up" from the text and display connections in a model view.	This is important for analyzing a case in depth instead of across cases and for getting the big picture of a project.
Making reports of data, codes, coding, etc.	Programs vary in the way reports are created and presented.	If you have particular report needs, check that these are supported.