Methodological Innovations Initiative

"Help, I'm getting buried in field notes!"

A Manual for Qualitative Data Management and Analysis

Rebecca Widom

with

Herbert Collado

MDRC

Manpower Demonstration
Research Corporation

January 2003

Contents

List of Tables, Figures, and Boxes

Table

Figure

Box

Acknowledgments

Thanks first to the Pew Charitable Trusts for the funding to compile this manual, provided through the Methodological Innovations Initiative.

Thanks also to Lyn Richards, of QSR International, for her careful reading of the manual and enthusiastic support for this work.

Finally, the contributions of colleagues from the Qualitative Data Management training series were indispensable. The manual benefited immeasurably from their feedback on earlier drafts, thoughtful questions, and in-depth project experience. In particular, Lisa Grossman and Linda Kato answered many questions about their experiences working on Opening Doors and Jobs-Plus. Barbara Fink, Laura Nelson, Janet Quint, and David Seith provided early feedback on the analytic approach. Throughout the process, Tom Brock gave critical support for this effort: suggestions on the substance of the manual and strategies for making it more user-friendly, referrals to additional literature and resources, and vital enthusiasm for rigorous qualitative research.

The Author

Key Recommendations for Managing Qualitative Data

A. Introduction

1. N5 is just software. It's pretty cool, but not miraculous, and won't do the thinking for you.

2. Use N5 if:

    a. You're using *narrative data* (interview transcripts, observational field notes, publications, etc.);

    b. You have *lots of data* (more than you can keep track of in your head);

    c. Your *budget and work-plan* assume that you will analyze the data, not just "talk to people and write a summary;" ***and***

    d. Your *data are high in quality* — really worth reading and re-reading.

B. What's the Plan? Qualitative Data Analysis and Coding

1. The goal is to build on the work that we already do (or would do anyway) and use N5 to systematize it and get more out of it down the road, not to revolutionize how we analyze qualitative data.

2. Do simple coding.

    a. Code very basic information about each document like site, date of data collection, and type of data (e.g. interview or observation).

    b. If you use semi-structured data collection methods, you are probably already informally coding your data by topic when you write it up. Using a template for written notes makes it easier to formalize this work so that N5 can automatically file the data by topic. Just remember to keep it simple — five or six broad topic headings, not thirty.

3. Don't over-code. This can't be emphasized enough. Pretend you only have a certain number of piles on your floor or highlighter colors, even if you're using N5. Your first draft list of codes will almost always be too detailed and too long.

    a. It's better to decide not to code for a particular concept, and then go back and look for that kind of text when you find that you really need it, than to spend time and money coding something that you'll never use. Over-coding is a big waste of time and money.

b. Over-coding — coding in too detailed a way — is also bad methodologically, because it tends to remove text from context and reduce our ability to get a full sense of what's going on. It's much better to focus on broad topics that are the most important, code them consistently and accurately, and allow yourself a chance to really read and think through the text related to those topics.

4. Don't over-delegate.

   a. Basic coding (like "site" or based on topic headings already in the notes) does not require much familiarity with the research questions or context from which the data are drawn. Delegate this.

   b. Substantive topic coding and identifying themes require consistent interpretive work and understanding of the research questions. Do not delegate this work to people who are not centrally involved in data collection and/or writing the report. If more than one person will be doing this type of coding, plan extra time for training, coding meetings, and inter-rater reliability checks. Even when there is a team of coders, the process may go more smoothly if there is one person ultimately responsible for deciding controversial issues.

5. Do balance staff time: If a person's time isn't concentrated enough on coding, then they will need to spend a much larger proportion of their time in training, but after several hours straight of this very detailed interpretive work, staying focused can become quite difficult.

6. Do budget for qualitative data management and analysis. At least budget for a half-time research assistant or analyst to keep track of the data files, import them into N5, and do very basic sorting (a.k.a. filing, coding).

C. Getting Started: Formatting Documents for N5

1. Table KR-1 translates N5 terms to non-computer based tools for analyzing qualitative data.

Methodological Innovations Initiative

Table KR- 1

Basic Elements of an N5 Project

What you use(d) without (before) N5	**The analogous element of an N5 project**
Written materials (data)	Documents
Codes or categories - highlighter colors - manila folders - piles on the floor	Nodes
Analytic notes - margin notes - summaries	Memos

2. Analytic Decisions in Document Preparation

   a. *Retaining meaning from original materials:* N5 deals with text only, so formatting in Word won't make it into N5. The chapter contains specific suggestions for keeping emphasis and other meaning contained in formatting when preparing documents for N5.

   b. *Preparing for sorting:* Templates for notes help staff to use keywords consistently in document and section headings, so that N5 will be able to automatically code documents and sections of text. Questions to keep in mind when deciding what kinds of keyword coding you want to use:

      - How would you analyze the data without N5?

      - What are two or three key basic characteristics of each document that would help organize the written materials?

      - How structured are the written materials? Are they already organized by topic?

   c. *Preparing for counting:* Not every project uses counting as a central analytic strategy; still, we often end up using counts in our writing, reporting how many people we interviewed or what "most" people said. N5 can count documents and text units. So, if you know early on what you may want to count, it's best to set up your project so that documents and/or text units correspond to the units that you want to count. The goal here is to use N5 to facilitate fact-checking, without changing the focus of the analysis.

D. Telling N5 What You Think: The Mechanics of Coding

1. Creating nodes is like labeling a manila folder — saying, “I’m going to put everything from Cleveland here.” The chapter describes a couple of ways to create nodes.

   a. N5 has several types of nodes. For our purposes, the text search, index search, free nodes (non-hierarchical) and tree nodes (hierarchical) distinctions are mostly meaningless. Just keep in mind that it’s always best to use a hierarchical node structure[1] when you have a regular categorical variable, such as “site,” as the parent and you would want to have child nodes “Cleveland,” “Los Angeles,” and so forth.

2. Coding is like doing the actual filing, putting the interview notes from Cleveland in the Cleveland folder. The chapter describes several ways to do that:

   a. Coding by hand, when you want to read the text on screen and code as you go.

   b. Using a coding template in Excel, when coders want to work from printed reports on paper without creating extra data entry work for data managers.

   c. Using “auto-coding” when you have keyword coded documents.

   d. Importing tables from Excel when you have a list of document names and basic information.

E. The Fun Part: Creating Output/Reports

1. The big caveat: This is how you quickly and easily get fabulous information back from N5! This is where all of the thinking and detail work in the steps above really pay off. Just remember that the information you get will be only as fabulous as the work you’ve done up to this point.

2. Types of reports: There are node reports and document reports. Remember that documents are like word files, such as your notes on a particular interview; nodes are like file folders, or a particular color of highlighter. If you’re starting to read through the data, you will probably want to read whole documents at a time, and you can use document reports for that. Later, rather than having to search through a big stack of field notes to find everything that you highlighted with a green marker, you can just click on the node that means whatever “green highlighter” would have meant, and click the “report” button to get a node report — a list of all of the text on that topic or from that type of document.

3. Searches:

[1] I sometimes call them “vector nodes,” but in the official N5 language, there is no such thing as a vector node, there is only a vector search.

a. Node searches rely on coding that you've already done, and allow you to access certain sections of text very easily once the coding is done. The chapter includes instructions for getting answers to questions like: What did caseworkers in Cleveland say about sanctioning? Give me the answers to question 5 by site. How many staff at each level did we interview at each site?

b. Text searches can be handy if you don't have nodes referring to what you want, and you know exactly what you're looking for (as in, if you're checking a quote) and/or are very familiar with the kinds of language used in the data. Cautionary note: don't forget the spell-checker poem (Appendix A), N5 will look for *exactly* what you tell it to look for.

4. Tips for fact-checking: If your N5 project is well documented and the paper refers to specific documents, text units, and/or nodes in fact-checking notes, it makes fact-checking *much* easier.

F. Sharing Data on a Team

1. The chapter discusses different models for sharing data from a highly centralized version with only one person having access to the N5 project to a decentralized model with off-site consultants and shared responsibility for analysis.

2. While not every project team will be able to use the most centralized model, the general guideline to keep in mind is "keep it simple." Beyond that, here are a few specifics:

   a. Always have one data manager (research assistant or analyst) who knows where the data are, brings documents into N5, logs basic information about the documents received, and keeps track of the versions of the N5 project whether there is just one or many. Note that this is probably at least a half-time job.

   b. Keep the number of versions of the project floating around to a minimum, so that you don't lose track of any documents or coding work.

   c. Consistent documentation is key. Keep lists of incoming data and where they're stored; coding decisions; versions of the project, where they're stored, and how they're named; and where all of the documentation is stored.

   d. Keep your data (the N5 project, raw data files, and project documentation) on the MDRC corporate network server.[2]

3. The chapter describes how to use WinZip to e-mail N5 projects to team members and QSR Merge if you need to merge projects from multiple team members.

[2] MDRC has a corporate network which is backed up daily and allows staff to share data. Because this manual was originally written for MDRC staff, "the network" refers to MDRC's network. Universities and other research settings often have servers maintained by central computer staff that can also be used to share data on a team.

G. N5 Cheat Sheet: A Quick Reminder for Some Common Tasks

Import a Document (Chapter III)

1. Step 1 in Word: Save the Word file as text.
   a. Open the file in word.
   b. Remove fancy formatting.
   c. On the menu choose File > Save As,
      - Remember to choose "Text Only (*.txt)" under the pull-down box labeled "Save as type."
   d. Close the file in Word.
2. Step 2 in N5: Import the text file into N5.
   a. Click "Import Documents" from the Project Pad.
   b. Select the text file version of your document (ends with .txt) and click the "OK" button.

Create a node (Chapter IV)

1. In the "Node Explorer" window, select the parent of the node you want to create — "Free Nodes," "Tree Nodes," or an existing node in the index tree.
2. Right click on the parent node and choose "Make free node," "Make top-level Tree Node," or "Make new Tree Node" from the menu that appears.
3. Enter the name for the new node in the node explorer window and click "Accept Changes."

Code with the Browser (Chapter IV)

1. Click on the document or node whose text you want to read and code, and click on the "browse" button in the document or node explorer window.
2. Click on the node you want to file text under in the node explorer or select it from the "recently used" list at the bottom of the browser.
3. Select some text in the browser window.
4. Click the "Code" button at the bottom of the browser window.

Searches (Chapter V)

1. Node Search
   a. Use this to search for combinations of nodes (e.g. caseworkers from Cleveland)
   b. In the project pad, under "Nodes," click on the button to "Search and Compare Nodes"
   c. The chapter includes more information on different kinds of node searches.
2. Text Search
   a. Use this to search for a specific quote
   b. In the project pad, under "Documents," click on the button to "Search Text of Documents"

Create a Report (Chapter V)

1. Creating a basic document report:
   a. Click on the document in the document explorer window.
   b. Click on the Report button.
   c. Select "Document Text" and "No Coding."
2. Creating a node report
   a. Click on the node in the node explorer window.
   b. Click the "Report" button.
   c. Select "Show Coding Data" and "Coded text."
3. Save the report and print it from KEDIT
   a. When you go to close the report window it will ask if you want to save the report, so say yes.
   b. Open the report file in KEDIT.
   c. Click on the button at the bottom of the screen that says "4P".
   d. A box will open asking you for optional commands. Click "ok."

E-mail the project (Chapter VI)

1. Start WinZip, if you don't see "Welcome to WinZip Wizard" then choose File > Wizard.
2. To send a project
   a. Start WinZip, and click the "Next" to start.
   b. Choose "Create a new zip file" and click next.
   c. WinZip will ask where to save the file, and this will be the file that you e-mail.
   d. Click the "Add folders" button and select the N5 project parent directory. (The folder that contains the project.stp file and all of the N5 subdirectories, but not the files themselves.)
   e. Click "Zip Now" and exit WinZip.
   f. In GroupWise, you can attached the .zip file you created above in the same way that you would attached a word file or anything else to an e-mail.
3. To receive/unzip the project
   a. From GroupWise, double click on the zip-file icon in your e-mail.
   b. Click on the "Next" button, choose "Unzip or install from [name of the .zip file in the e-mail]," and click "Next" again.
   c. Click on the "Select different folder..." button and choose where you want the N5 project to be saved.
   d. Click "Unzip Now."
   e. Click the "Close" button to exit WinZip.

I. Introduction

A. How Do I Use This Manual?

1. This manual is intended for a very broad audience: from senior researchers managing projects that use qualitative data to assistants keeping track of field notes for a particular project; those with years of experience collecting and analyzing qualitative data and those who have never even read an ethnographic study; those whose job responsibilities include computer programming and those who have experienced a wide range of difficulties and discomfort with computers.

2. Understanding how the manual is organized will help you find the information you need:

   a. The manual starts with more conceptual, methodological, and management issues, and moves into the specifics of using N5 later. This is the overall flow of the manual, and also the particular flow within chapters.

   b. The manual roughly follows the process of qualitative data management and analysis, beginning with decisions about whether or not to use N5 and the design of the study and moving through using N5 to create output and fact-check reports.

3. If you are interested in a broad overview, you can just read the Key Recommendations and the first two chapters. If you are interested in how to do a particular task in N5, start with the N5 Cheat Sheet on the previous page, and check out the end of the chapter describing the step you're working on for more information.

B. Caveats

1. Most of this manual focuses on advance planning and organizational work so that you don't end up buried in field notes. If you're already buried, there's no magic solution here — you just have to sort through your notes.

2. Compared with the breadth and depth of scholarly work on qualitative research methods, this manual has an extremely narrow focus, concerning only methods of qualitative data management and analysis and types of qualitative data in use at MDRC.

3. There are a number of crucial skills for qualitative research *not* covered in this manual, but discussed at length in the literature on qualitative research methods in social science, to list just a few: access to sites of study and key informants, sampling, interviewing and observation skills, writing up field notes, and writing up findings based on qualitative data.

4. For some more information on qualitative research methods relevant to work at MDRC, see

   a. Lofland and Lofland (1995) on qualitative field methods, from data collection to analysis and writing.

b. Glaser and Strauss (1967) on grounded theory, a central approach to qualitative research.

c. Emerson, Fretz, and Shaw (1995) on field notes — a practical guide to recording what we observe.

d. Miles and Huberman (1994) on strategies for analyzing qualitative data — a classic.

5. As with much writing about software, much of this manual will be almost outdated by the time that it's published. N6 has already been released and this manual focuses on N5. As projects continue to use N5 software there is a constant stream of innovation building new tools that will soon seem indispensable. Moving forward, the MDRC intranet will be a key repository for new tools and information.

C. Overview of Qualitative Research at MDRC

1. What does qualitative data analysis usually look like at MDRC?

   a. Descriptive Analyses.

   - What happened when this policy went into place?
   - How were services delivered?
   - How did staff and clients experience this program?
   - What does this statistic look like "in real life"?
   - What is the context in which the program operated?
   - What are some promising practices for future study?

   b. These analyses, in turn, are often used to answer further questions:

   - Did the program receive a "fair test"?
   - How do we explain the impacts?

   c. All of these purposes lend themselves to a fairly atomized, topic-focused, analytic strategy (Willis and Jost, 1999).

2. Three MDRC projects are used throughout the manual as illustrations: Urban Change Implementation (Brock et al., 2002), Jobs-Plus (Kato and Riccio, 2001), and Opening Doors (Grossman et al., 2002). These projects are diverse in their methodology and subject matter, and represent most of the breadth of approaches currently in use at MDRC.

Methodological Innovations Initiative

Table 1

Summary Information about Three Sample Projects

	Opening Doors: Focus Groups	**Jobs-Plus: Building New Partnerships**	**Urban Change Implementation: Welfare to Work Services**
Brief description	Focuses on improving access to community colleges and credentials for non-traditional students.	Implementation report, looking at the formation of Jobs-Plus Partnerships in each site.	Research in the Urban Change Implementation Study, looking at Welfare-to-Work services in four big cities.
Research subjects	Three groups — current community college students, former students, and potential students (who had not pursued higher education)	Partnership participants	Line staff and administrators in neighborhood welfare offices in each city
Data collection method	Unstructured focus groups and structured phone interviews (not in N5).	Interviews, e-mail correspondence, participant observation, annual reports, newspaper articles, agency documents	Semi-structured interviews
Data format	Transcripts (for focus groups)	Primarily field notes, some keyword coded and others not	Field notes, some keyword coded and some not
Staffing data collection	Lead consultant(s) conducted focus groups with MDRC staff present. Transcription company typed transcripts. MDRC staff reviewed the tapes and transcripts for accuracy.	Consultants in each city were primarily responsible for data collection in their location. MDRC staff attended some visits as well, and had frequent contact with site researchers.	MDRC staff conducted interviews and wrote up field notes.
Staffing analysis	MDRC staff and consultants collaborated on analysis. There were meetings with consultants to agree on and define codes. Research Assistants (at MDRC and Virginia Tech) did initial coding, and had ongoing training in meaning of codes.	Initial coding of a first round of data conducted by a research assistant and used by one author, but not the other. Most analysis conducted without NUD•IST.	MDRC staff conducted analysis. Initial filing done by research assistants, who then printed reports from certain interview questions by site for authors (associates and senior associates) to use in writing their sections.

D. Understanding What N5 Can Do and Deciding Whether or Not to Use It

1. When should I use N5?

   a. Figure 1 outlines key decision points regarding using N5, and describes some alternative methods for qualitative data management and analysis.

2. What will I gain by planning for and using N5?

a. Analysis is easier if all of the documents are in one place and easily accessible over time and across staff turnover. The software alone won’t make this happen, but it does provide a central storage system.

b. N5 dramatically improves information retrieval by replacing color-coded tape flags and highlighters with electronic folders like those used in Windows.

c. N5 makes fact-checking easier, because it makes it easier to provide other people with all of the data and documentation of your analysis.

d. N5 helps you answer these kinds of questions, *once you have set up the project and done the coding*:

- What did caseworkers say about sanctioning?
- Give me the sanctioning information by site.
- Which levels of staff did we interview at each site? And how many interviews in each group?
- How many teachers said that they thought the new curriculum was a good idea?
- In how many sites did they implement time limits?
- Give me all of the interview text including mentions of clients’ “attitudes.”

3. What not to expect from N5

a. The purpose of N5 is easy retrieval of information. It will not do the organizing or the thinking for you. It works best when building on strategies you’re already using.

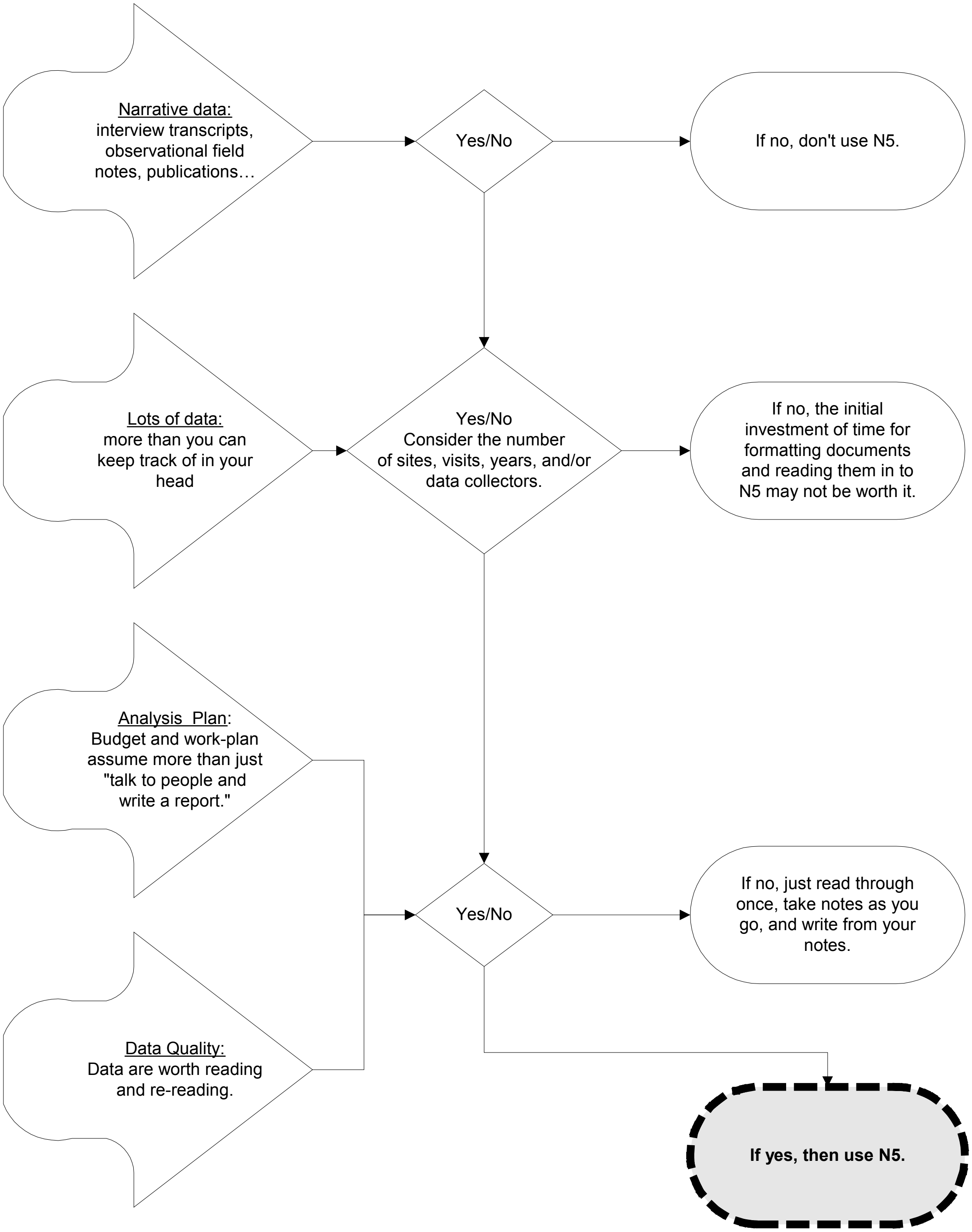
b. Just like using Word doesn’t necessarily make your writing better or faster, using N5 won’t necessarily make your analysis of qualitative data better or faster. It’s just software.

4. N5 is pretty cool, but not miraculous. There is more work upfront, in exchange for more convenience and efficiency later on.

Starting at the beginning you will need to implement decisions to make answering these questions easier at the end. And, as with research design in general, setting things up with one set of questions in mind at the beginning, may make it more difficult to answer a different set of questions at the end.

Figure 1

Deciding Whether or Not to Use N5


Narrative data:
interview transcripts, observational field notes, publications…
Yes/No
If no, don't use N5.
Lots of data:
more than you can keep track of in your head
Yes/No
Consider the number of sites, visits, years, and/or data collectors.
If no, the initial investment of time for formatting documents and reading them in to N5 may not be worth it.
Analysis Plan:
Budget and work-plan assume more than just "talk to people and write a report."
Data Quality:
Data are worth reading and re-reading.
Yes/No
If no, just read through once, take notes as you go, and write from your notes.
If yes, then use N5.

II. Qualitative Data Analysis and Coding

A. Analyzing Qualitative Data at MDRC

1. Research questions

   a. Rather than developing theories of human behavior, we most often describe what happened in a particular context, along dimensions that policy-makers and other researchers expect to contribute to program impacts. In some studies, we have also used qualitative data to illustrate, amplify, or qualify key points either from quantitative research, as in Polit et al. (2001), or from discussions with funders and policy makers, as in Grossman et al. (2002).

   b. We also use qualitative data to answer further, more comparative/analytic, questions: Did the program receive a fair test? How do we explain the impacts?

   c. There are other ways of analyzing qualitative data, but the methods presented here work well for the kinds of research we do at MDRC.

2. Challenging characteristics of our work:

   a. *Large multisite, multiyear studies:* This means that there are often a lot of data. For example, Jobs-Plus has two five drawer file cabinets of written notes, and the Opening Doors project analyzed about 1400-1500 pages of transcripts.

   b. *Large teams of researchers with staffing change over the course of the project:* The traditional ethnographic model involves one person doing all of the data collection, analysis, and writing, but that usually doesn't happen here. There are advantages and disadvantages to this approach. Everyone is different and any section of text can be interpreted in myriad ways. This means that multiple people doing the same interpretive work will probably be less consistent than one person doing all of the work. However, coming to agreement — filtering interpretations through more than one person's perspective — generally leaves you with more trustworthy interpretations.

   c. Limited time and budget

B. Overview of the Process of Qualitative Research

1. Note: This process doesn't fully apply to archival or historical research, where the raw data are texts written by others, but we tend not to do much of that kind of research at MDRC.

2. In general, this is the process of going from a collection of specific interactions (interviews, math classes, focus groups, job clubs, etc.) to a more broad description of "what happened" or "what we learned." The process involves distilling a much shorter written report from pages and pages of detailed narrative. Generally speaking there are the following steps, which happen roughly in the order below, but with plenty of overlap and backtracking in many cases.

   a. *The "raw" data are interactions:* Throughout the study, we should consider how those interactions are selected (study design and sampling) and structured (interviewing skills, participants' motives, etc.).

   b. *Represent the interactions in written form:* In Opening Doors, focus groups were taped, and transcriptions were created and checked against the tapes and other written notes. In Urban Change, MDRC staff went to the sites and conducted interviews and observations which they then wrote notes about. There is a lot to say about how best to accomplish this step, but this manual focuses exclusively on how the resulting texts are structured (or not), and how that relates to the other steps.

   c. *Gather all of the written materials together:* Teams have used different methods for keeping track of notes written by different people over several years, and depending on how much up front organization has been done, this step can take more or less time. MDRC seems to be moving towards having a central directory on the network[3] and/or intranet where team members store all written materials as they are created, and as long as the full team is consistent in this, gathering the data for analysis won't be so complicated.

   d. *Get some basic familiarity with the data and decide on key topics of interest:* This can mean formally reading through everything once or twice, keeping notes as you go, and distilling a list of key topics from the notes. What seems to happen more often at MDRC, is that at least one person doing the analysis was also centrally involved in collecting the data, read materials collected by other team members as they were collected over time, and has enough direction from their colleagues to know the key topics of interest.[4] The former is preferable in that it relies less on individual memory of the interactions and more on the recorded data, but it also takes more time, and to the extent that people feel certain in advance about what they're looking for, it may not be necessary.

   e. *Organize the data by topic:* This step could take a number of forms: different colored high lighters, physically cutting up pieces of paper and putting them in different manila

[3] MDRC has a corporate network which is backed up daily and allows staff to share data. Because this manual was originally written for MDRC staff, "the network" refers to MDRC's network. Universities and other research settings often have servers maintained by central computer staff that can also be used to share data on a team.

[4] Sometimes, there may also be an opportunity to go and collect further data to fill in gaps on some topics.

folders, cutting and pasting electronically in MS Word, making piles of paper spread across the office floor, or coding text in N5. Some of this work can be fairly simple (e.g. filing notes by site or when the data were collected), some involves more interpretive work (e.g. finding the focus group conversations about "family"). When complete, this step allows you to access, for example, all of the information on sanctioning (for the welfare to work paper), resident participation (in Jobs-Plus), or financial aid (in Opening Doors).

f. *Read the data on each topic looking for themes:* Analysts read through subsets of the data, often more than once, to both identify and confirm emerging themes. There is some intuitive work in identifying important ideas, and ongoing comparison of the different texts and clarification of each idea, to see if the idea represents a preponderance of the data. For example, now that I have access to all of the information about financial aid, what kinds of things are focus group respondents saying about it? Do they seem to be well informed about their options? What counts as "well-informed" in this context? This step can take a number of forms: it may look like cutting and pasting interesting quotes from electronic notes into word, taking further margin notes, creating lists with different potential themes and making notes of where they appear, and/or additional coding in N5.

g. *Write the report:* Although we won't focus on writing qualitative research here, writing is the final step of analyzing qualitative data.

C. Guidelines for Analyzing Qualitative Data with N5

1. The following suggestions, based on MDRC's project experience, are organized by steps in the process with some general comments on management at the end.

2. If possible, do simple keyword coding when writing up your notes. If you know enough about the analysis plan in advance and/or if you use semi-structured data collection methods, it's a very good idea to include consistent keyword coding in writing up your notes. This can and should be done in a way that merely formalizes coding already taking place as part of data collection. In other words, think about the document and subject headings you and your team would have used in writing up your notes anyway (*if any*) and just make sure that everyone uses those same headings consistently.

   a. *What is keyword coding?* It's inserting consistent keywords to mark off sections of text for a particular code. (See Box 1 for an example and Chapter III for more details on the mechanics.)

   b. *Do use a template for writing up keyword coded notes:* For auto-coding to work, each of the sub-headings needs to be spelled exactly the same way, and templates can help a lot with this.

- In Urban Change, for the more structured interviews with case managers, the write-ups also followed the main topics in the interview guide. The team developed a template for notes in MS Word, which was designed to include standard keywords (e.g. <Q1> for question 1) that N5 could search for and use to automatically code sections of text. However, the team was not fully consistent in using the template and there were some quirks with Word's automatic numbering that made it more challenging to use consistently, so auto-coding did not work for this project.

- Text Box 1 gives an example of some keyword coded text from a Jobs-Plus template. Note that keywords are spelled exactly the same in each document (i.e. *ChComp1 not *Ch-Comp1).

c. *Don't use complex keyword coding,* a corollary of "don't over-code" below. If you come up with 100 questions for data collectors to answer that they wouldn't have answered otherwise, either a) they won't do it or b) they will understand at least some of the questions in ways that you didn't intend, making the resulting coding less reliable.

- For example, on Jobs-Plus, MDRC staff developed a "template" for the on-site researchers to use in submitting their notes for the Building New Partnerships report. The template included over 100 keyword-coded questions, some of which could be used as semi-structured interview guides. This allowed on-site researchers to do topic coding, without having to use NUD•IST, and the MDRC team was able to quickly and cheaply auto-code the templates in NUD•IST.

- Unfortunately, the on-site researchers found the template very difficult to work with because the questions (topic codes) were too detailed, and their perceptions of what had happened in each of their sites did not fit well into these categories. MDRC staff developed a checklist, which they used to evaluate the quality of data submitted in this format and ultimately found that the quality of data varied dramatically by site. The original goal was to receive good summaries of the original data, but some site researchers sent transcripts, some sent summaries without any quotations at all, others sent both transcripts and summaries, some write ups contained the same quotes in different questions, etc. Research assistants at MDRC ended up having to code some of the original data from the sites with more troublesome templates.

Box 1

Keyword Coded Text from Jobs-Plus

Baltimore:

*ChComp, Changes in Composition of the Collaborative

*ChComp1, which partners were added and why?

During the original application period, the Mayor's Office of Baltimore City pulled together a group of organizations for the collaborative. After discussions with MDRC regarding the goals of Jobs-Plus, more service providers became involved and the policy makers became less involved. Danise Dorsey Jones who was involved in the process stated that "We looked at who is the customer and what agencies touch the customer. So for instance, the housing authority as a stand-alone non-profit had to be at the table.

<...additional text...>

*ChComp2, which partners left and why?

Two partners listed in the original grant, Department of Recreation and Parks, and the Department of Education are not actively involved in the collaborative (B_a2, Page 5-6).

<...additional text...>

Cleveland:

*ChComp, Changes in Composition of the Collaborative

*ChComp1, which partners were added and why?

"Yes, new members were added to the collaborative. They were added as a gap was identified, or based on the actual need,or really the level of need in the community." (Persse interview with Peter W., April 8, 1999)."

An example of how these decision were made can be seen in how one of the agencies that has assumed a key role in Jobs Plus was added. Vocational Guidance Services is a long-"established agency that focuses on job preparation, training" and placement for people with disabling conditions/barriers to work.

<...additional text...>

*ChComp2, which partners left and why?

Imani Community Action Partners (ICAP) was actively involved in Jobs-Plus from summer/fall 1997 through summer 1998 and its exit from the initiative was an event that slowed the planning process. The agency expected to receive a contract from CMHA that would have provided salary for 1 full-time staff person and several part-time people who would perform coordinating and case management duties for Jobs-Plus and be housed at Woodhill..

<...additional text...>

d. *Don't do keyword coding just to do it.* If the study is designed in a more inductive or exploratory way — that is if you don't start data collection with a definite sense of the topics that will be important — then keyword coding will not be worth the effort.

- For example, on Opening Doors, team members checked the transcripts against narrative notes, and then inserted keywords based on the questions in the focus group interview guide. However, because the guide was not rigidly followed in the focus groups (because Opening Doors was an exploratory study) these categories were not terribly helpful for future steps in the analysis.

3. Keep all of your data in a central location

a. This is always easier to do sooner rather than later, when staff who collected the data have left MDRC and/or misplaced their original notes.

b. It can be very helpful to place one person on the team in charge of collection and organizing the written materials as the study progresses. All of the sample projects had one or two people in charge of keeping track of the data and this worked well for them. For example, on Urban Change,

- Data collectors (approximately six people on the Urban Change implementation team) sent notes to a research assistant in charge of keeping track of them. She then logged and stored electronic copies centrally on the network, as the data came in.

- This process worked well. The only problem was that one of the authors was in the California office and could not directly access these files, but she could get them through the research assistant.

c. This is crucial whether or not the team will eventually be using N5. (See Chapter I for more information on deciding whether or not to use N5.)

4. Always do some basic coding.

a. Some basic information — the site, the date of the interaction, maybe the type of data and/or respondent (e.g. staff level in Urban Change, or type of focus group in Opening Doors) — doesn't require a great deal of interpretive work and should *always* be coded.

b. Basic coding also includes using subheadings in the documents to code text on particular topics. Again, this only works for semi-structured data.

- For example, in Urban Change, where the templates didn't work for auto-coding, research assistants were able to do basic topic coding following the document subheadings and semi-structured interview guide. This process was faster and cheaper than having senior associates do this coding. Had the templates worked properly, the process could likely have been completed in a week with auto-coding, but that might have taken slightly more time from those collecting the data.

5. Substantive topic coding requires consistent interpretive work and understanding of the research questions. As such, it takes more time and requires more expertise.

    a. Introduction to substantive topic coding

        - Substantive coding is like basic coding, except that there are no clear road signs like document headings or sub-headings. You just have to read the text, and decide which topic(s) it should be filed under.

        - Text Box 2, includes some text from one of the focus groups in the Opening Doors project. One example of substantive coding, would be to look at this text and identify all of the line numbers that reference text about social (not financial) supports for attending school. Another would be to identify the text focused on time management. (Note: If you and a colleague both try coding this text to these categories without talking about it in advance, you'll likely select different line numbers.)

        - How did the Opening Doors Team do it?

            – To begin topic coding, the team held meetings to brainstorm a list of 35 key topic areas, which were revised and refined during the process of training and coding. Approximately six people read through the transcripts (1400-1500 pages in total) to code these categories. There was an initial training where everyone coded the same transcript and compared results, and ongoing questions were resolved in regular team meetings.

            – Working with such a large group did not save nearly as much time as one might imagine, because of time needed for training. The two co-authors didn't always agree on coding decisions, so with four additional perspectives, the interpretations were that much more diverse. (Note: There is a SAS macro that can work with data exported from N5 to perform formal statistical tests of inter-rater reliability (Li and Lautenschlager, 1999).)

Box 2

Sample Focus Group Text to Code

and I knew I could do. And a lot of people had told me I couldn't do it, and that was something that's always good. Whenever somebody says I can't do it, I get to prove them wrong and say I'm going to do it. And that was one thing I had that helped me out.

MODERATOR LISA: Did you take like one course at a time or--

L: No, I went in four classes. Yeah, I had 12 credits, so it was Tuesdays and Thursday nights from 5:00 until 10.

MODERATOR LISA: What about the rest of you? How was it juggling all of that?

W: It was frustrating for me. When I went back, I was at somewhat at a low in my life and I was trying to get back. And I was I think like 25, 26. I was in a class with a lot of younger students. And at that time, I did, I took too many classes. I didn't have any idea what I should have been -- how to alternate my classes as far as the heavy loads and the less loads. So, my first two or three semesters were heavy loads, 12 credits with all of my electives, all of what I had to have. And that was really, really hard because it put like a damper on my spirits because I wasn't able to perform, you know, to the fullest in each subject because I was so bogged down with something else. And these were classes that I had to do good in, you know, and that I had to have. Had I known, you know, how to take my classes and stuff, I wouldn't have had such a hard time, you know, with my grades and the homeworks and different things like that.

The frustration is really what I believe led to me just dropping it and not going back on it. That was basically my thing, the frustration.

N: My problem ran with time management and I ran out of finances. So, what I did was I tried to get creative and I think that if I would have just stuck it out, I probably would have finished my schooling. I got up, I ran a day care out of my home and then I split college and work in the evening. I waitressed at night because I could make more dollar money in a shorter period of time, and go to college the other days like I do, three days of college and three days -- I know I filled every day. It was kind of like half and half was my nights I did college and work, and then I ran a day care during the day to pay the bills because it cost me money (A) to afford school, (B) for my kids because I was in school, someone else to watch them. I even got really creative and asked the neighbor downstairs to swap me back and forth. I'd take her kids in my day care for no cost if she'd take my kids when I was at school. And I just burnt myself right out. I had a little girl that had seizures, so it just kind of put an absolute halt to everything I was doing and I just went -- I mean I think I even left work and just did nothing for two months, went to my mother's and tried to figure out what to do with my little girl, she had seizures.

But if you're really creative with money and time and everything and you're super-woman, you might be able to do it. If I had a husband, it might have helped a little bit, but doing it myself it was really hard.

- It's much easier to go through a subset of the text and reject things that don't really fit, than to re-read the entire dataset to look for other text that should have been included. So, the team decided that the general coding rule was "If you're not sure, include the text." Still, the authors felt frustrated later reading through text that seemed irrelevant in reports from NUD•IST. For example, in the code "family," sometimes any reference at all — such as "yes I have kids" — was filed under that topic, rather than only more specific text about how family relationships informed community college experiences.

b. Substantive coding is much easier if the people analyzing the data were also centrally involved in collecting them and will be writing the report.

- *If more than one person will be doing this type of coding, plan extra time for training, coding meetings, and inter-rater reliability checks.* If one person does this coding in a relatively short period of time, we can more easily assume that their interpretations are fairly consistent. And, this is something that can be communicated, but with any given category or code (financial aid, for instance), there can easily be ten or twenty different distinctions: Does this refer to people saying that they need more money, or only to information about their experiences with formal financial aid programs? What about someone talking about a friend of a friend getting a Pell Grant, but not anyone she knew directly? What about someone saying that they've heard that the government used to give people money to go to college in the 1970s, but it doesn't anymore? Or making a sarcastic comment about government financial aid?

6. Don't over-code. This cannot be overemphasized.

a. The purpose of N5 is easy retrieval of information. Over-coding minimizes the text you have to read in the future, but also leaves you with less information. It increases the likelihood that data will be taken out of context, or that you will miss important related issues that were coded differently.

b. Before there was NUD•IST, researchers were constrained by the number of categories that they could keep in their heads, margin notes, post-it flags, or highlighters. With N5 you have none of those technological restrictions, and so it can be tempting to create a multitude of substantive interpretive categories, but it's still better not to do that.

c. Coding unstructured data, especially substantive coding, is expensive and time consuming, so it's best to only code what you absolutely know you will use — only label file folders that you know you will want to read through later. If your data is semi-structured, then use the existing structure. If not, prioritize a few broad categories that you can use to look at text in more depth later.

7. Managing Data Management:

   a. *Do balance staff time:* There is a bit of a rhythm to coding: if a person's time isn't concentrated enough on coding, then they will need to spend a much larger proportion of their time in training, but after several hours straight of this very detailed interpretive work, staying focused can become quite difficult. So, for example, rather than 20% of 5 people's time or 100% of one person's time for six weeks, I would recommend either two people at 50% each for six weeks or one person coding 75% of the time and doing other project work 25% of the time for eight weeks.

      - For example, on Opening Doors, there were about six people who worked on the analysis all for less than half of their time, and some for only part of the analytic process. Had there been two full-time people working on the project consistently throughout the process, it might have gone more quickly, but there were much smaller chunks of more people's time, which increased the need for training and management activity.

   b. Don't over-delegate.

      - Basic coding and data management (such as, getting N5 to file documents that have already been keyword coded) are skills that do not require much familiarity with the research questions or context from which the data are drawn. Delegate this.

      - Substantive topic coding and identifying themes are analysis. Do not delegate this work to people who are not centrally involved in data collection and/or writing.

   c. *Do put one person in charge of keeping track of the written materials*, even if it's just ensuring that everyone saves files to a particular directory on the network.

   d. *Do budget for qualitative data management and analysis*. When planning a budget consider: the number of pages of written materials, the number of people collecting the data, the structure of the data (from perfectly consistently keyword coded to completely free-form and inconsistently labeled), involvement of authors/analysts in study design and data collection, and the scope of the report envisioned.

      - On Jobs-Plus, Linda Kato, an Associate, took three months, working full-time to read the templates; plan, execute, and write up another round of visits; and draft her chapters of the report. No one knows exactly how much staff time was spent developing the template, auto-coding it, and checking the data within it, but two research assistants and one associate worked on this task at least part-time, and my sense is that it took at least a month.

- On Opening Doors, checking the transcriptions and conducting initial coding trainings took approximately one month, with two Full-Time Equivalents (FTE's) spread over three or four people at a time. The topic coding took an additional two months, at approximately two FTE's. Further formal coding for themes would have likely taken another three to four weeks of the co-authors' time.

e. *Do use N5 for fact-checking documentation.* N5 can be a very convenient central place for documenting qualitative analysis, and can greatly facilitate fact-checking qualitative data. If coding is accurate, consistent, and well documented, authors can simply e-mail an N5 project to the report coordinator rather than handing over stacks and stacks of papers with check lists and so forth.

III. Getting Started: Formatting Documents for N5

A. Basic Elements of an N5 Project

1. What do we have to work with?

   a. When you're talking about qualitative data you have the text itself and codes or categories and analytic notes or summary memos that you may use to organize the data in your mind.

   b. This translates into documents (which are divided into "text units" and can also be divided into sections), nodes (which are the codes/files), and memos (which I won't focus on as much here, but are a space for writing analytic or process notes).

Methodological Innovations Initiative

Table 2

Basic Elements of an N5 Project

Before/without N5	**In N5**
Written materials (data)	Documents
Codes or categories - highlighter colors - piles on the floor - summary check-lists	Nodes
Analytic notes - margin notes - summaries	Memos

B. Analytic Decisions in Document Preparation

1. Retaining meaning from original materials

   a. N5 is very flexible and can work with many types of data — field notes; transcripts; Smith and Short (2001) wrote an interesting piece about using it to refer directly to video or audio tape (bypassing transcription); in the Urban Change Institutional Study, we wrote up summaries about flyers and brochures which were appended to interview notes; etc.

b. See below for more details on keeping meaningful formatting (e.g. for added *emphasis*) in plain text files.

2. Preparing for sorting

a. The goal of sorting at this phase

- Whereas quantitative data comes pre-sorted into variables, qualitative data is less structured, and where to look for information on a particular topic or even which topics are addressed most frequently in the data is less obvious. Sorting allows us to arrange the text into categories, to impose some structure on the text in order to facilitate further analysis.

- During document preparation, you can implement keyword coding for automated filing (auto-coding).

- This facilitates access to certain sets of text later on.

- The goal is to make the most of analytic work already done during the process of converting the interaction to text.

b. All three sample projects used some keyword coding

- Jobs-Plus template has a section for each question and standard basic information in the header.

- Opening Doors inserted keyword codes into the transcripts for questions in the focus group guide.

- Urban Change used less consistent keyword coding for the site and type of notes and for more structured interview guide questions.

c. Sorting Questions

- How would you analyze the data without N5?

- What are two or three key basic characteristics of each document that would help organize the written materials?

  – UC: site, staff level, interview or observation

  – Opening Doors: site and type of focus group

    - Jobs-Plus: site, template data or not

- How structured are the written materials? Are they already organized by topic?

    - Jobs-Plus templates and more structured interviews in Urban Change, yes.

    - Opening Doors focus groups, no.

3. Preparing for Counting

    a. To count or not to count

    - Is counting central to your analysis?

    - What about reporting sample sizes?

    - What about "quantitative terms or modifiers"?

        - "All [17] of the Philadelphia case managers also recommended English as a Second Language training for a limited-English speaking recipient." (UC Implementation, Chapter 4 and Table A.1)

        - "Most of the current, former and potential students recognized the value of college education as a means of improving the financial well-being of their families." (Opening Doors, draft executive summary)

    b. What to count?

    - Structured surveys count survey respondents.

    - Qualitative analysis can count many different kinds of things, all from the same dataset:

        - monthly Jobs-Plus Partnership meetings

        - organizations participating in Jobs-Plus

        - people attending the meetings/representing organizations

        - public housing residents

    - You may not know what you want to count until you're well into the analysis.

- The goal is to use N5 to facilitate fact-checking without changing the focus of your analysis.

c. What can N5 count?

- In order to use N5 for fact-checking quantitative statements and approximations, it helps immensely to have your units of analysis represented by something that N5 can count. N5 can count documents and text units.
  - UC Implementation
    - 1 document = 1 interaction. Among those documents that are case manager interviews, each document represents one case manager interview.
    - If you think that you'll be wanting to make statements about the number of caseworkers who said such and such, and you did one or more individual interviews with each caseworker, it might make sense to have a document for each caseworker or each separate person interviewed, even though each document might contain more than one interview over time.
  - Opening Doors
    - 1 document = 1 focus group; 1 text unit = 1 "mention"; can also use cases
    - Each text unit is a particular statement from a particular person in the focus group. (For example, "Sally: I really loved my classes, they really helped me out on my job, and my professors were so nice, and I learned so much from the other people in the class, and I never was bored, and [blah blah blah]…" and "Joe: I didn't like mine." would each be a text unit.) This way, counting text units coded to a particular node provides a rough estimate of the number of times the corresponding topic was mentioned. Also, each document was a focus group, so the number of documents with text coded to a particular node tells you how many of the focus groups had some discussion of that particular topic, regardless or how brief or extensive the conversation was.
- N5 automatically counts "documents" and "text units"
  - UC Implementation: 1 doc = 1 interaction (interview, observation, etc.)
  - Opening Doors: 1 doc = 1 focus group; 1 text unit = 1 "mention"

    - Jobs-Plus: documents and text unit counts don’t mean much

  - N5 can also output information on “cases,” but that is more complicated

    - See *N5 Reference Guide*, pp.115-123

d. How does that work in practice?

  - If you know that you want to count and what you want to count in advance, then define documents and/or text units so that they represent what you want to count.

  - If you don’t know, don’t worry about it.

    - You can use “comments” in Word to make more detailed notes for fact-checking.

4. Anatomy of an N5 document

   a. Box 3 shows a sample document in Word, Box 4 shows it saved as text only, and Box 5 shows an N5 report of the document.

Box 3

Word Version of a Sample Document

* this text at the very top is the document description (no blank lines above it)
* all of the text units start with an asterisk
* you may want to include basic info: site, date of data collection, person who wrote the notes, etc. [this is still one text unit because I didn't hit <return>.]
* Unfortunately, you can't code text in the document description.

#these refer to document codes, with info that N5 can apply to the whole document
#site=New York City
#organization=MDRC
#author=Widom

This is some text. It won’t be included in any "section" because there is no section header above it. But, that's ok, sections aren't always that helpful. For example, you may not want to use section headers in observational notes that aren't easily divided into sections, as opposed to more structured interview notes where each section can be an interview question/topic.

Note that there has to be at least one blank line, document code line, or other line of text between the document description and the first section header. Otherwise, N5 will think that the first section header is part of the document description.

*First-Section-Header

here is some random text to go in the first section
here is another text unit in this section
this * (asterisk) doesn't start the next section, because it isn't the first character in the text unit.

*Second Section Header: it doesn't matter how much text you put in a section header. The only thing that matters is that the text unit starts with an * and you use **consistent** keywords for section headers if you want to use them for auto-coding.

Blah blah blah

*apostrophes and quotation marks

Word automatically formats your apostrophes, in a way that can’t be saved clearly in a text file. You may want to search for the curved apostrophe's and quotation marks and replace them with regular vertical ones in K-Edit. It’s not a huge big deal, because you get used to the weird characters, but it does make things somewhat easier to read in N5.

This is a quote: “The same goes for quotation marks.”
So is this: "But here I use the regular straight vertical quotation marks."

*One last word (section, really) on formatting

Note that there are no **bold** or *italics* or anything in this sample document. You can use **CAPS**, *+plus signs+,* or ***+BOTH+*** to indicate emphasis instead, because N5 can't read formatting.

Box 4

Text only version of the Sample Document (This is ready to be imported into N5)

* this text at the very top is the document description (no blank lines above it)
* all of the text units start with an asterisk
* you may want to include basic info: site, date of data collection, person who wrote the notes, etc. [this is still one text unit because I didn't hit <return>.]
* Unfortunately, you can't code text in the document description.

#these refer to document codes, with info that N5 can apply to the whole document
#site=New York City
#organization=MDRC
#author=Widom

THIS IS SOME TEXT. It won't be included in any "section" because there is no section header above it. But, that's ok, sections aren't always that helpful. For example, you may not want to use section headers in observational notes that aren't easily divided into sections, as opposed to more structured interview notes where each section can be an interview question/topic.

Note that there has to be at least one blank line, document code line, or other line of text between the document description and the first section header. Otherwise, N5 will think that the first section header is part of the document description.

*First-Section-Header

here is some random text to go in the first section
here is another text unit in this section
this * (asterisk) doesn't start the next section, because it isn't the first character in the text unit.

*Second Section Header: it doesn't matter how much text you put in a section header. The only thing that matters is that the text unit starts with an * and you use +consistent+ keywords for section headers if you want to use them for auto-coding.

Blah blah blah

*apostrophes and quotation marks

Word automatically formats your apostrophes, in a way that can't be saved clearly in a text file. You may want to search for the curved apostrophe's and quotation marks and replace them with regular vertical ones in K-Edit. It's not a huge big deal, because you get used to the weird characters, but it does make things somewhat easier to read in N5.

This is a quote: "The same goes for quotation marks."

So is this: "But here I use the regular straight vertical quotation marks."

*One last word (section, really) on formatting

Note that there are no bold or italics or anything in this sample document. You can use CAPS, +plus signs+, or +BOTH+ to indicate emphasis instead, because N5 can't read formatting.

Box 5

N5 Report on a Sample Document

```
QSR N5 Full version, revision 5.0.
Licensee: Rebecca Widom.

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++ ON-LINE DOCUMENT: sample-document
+++ Document Description:
* this text at the very top is the document description (no blank lines above it)
* all of the text units start with an asterisk
* you may want to include basic info: site, date of data collection, person who wrote the notes, etc. [this is still one
text unit because I didn't hit <return>.]
* Unfortunately, you can't code text in the document description.

+++ Retrieval for this document: 40 units out of 40, = 100%
++ Text units 1-40:
                                                                                  1
#these refer to document codes, with info that N5 can apply to the whole          2
document                                                                          3
#site=New York City                                                               4
#organization=MDRC                                                                5
#author=Widom                                                                     6
THIS IS SOME TEXT. It won’t be included in any "section" because there            7
is no section header above it. But, that's ok, sections aren't always             8
that helpful. For example, you may not want to use section headers in             9
observational notes that aren't easily divided into sections, as opposed         10
to more structured interview notes where each section can be an interview        11
question/topic.                                                                  12
Note that there has to be at least one blank line, document code line, or        13
other line of text between the document description and the first section        14
header. Otherwise, N5 will think that the first section header is part           15
of the document description.                                                     16

*First-Section-Header                                                            17
                                                                                 18
here is some random text to go in the first section                              19
here is another text unit in this section                                        20
this * (asterisk) doesn't start the next section, because it isn't the           21
first character in the text unit.                                                22

*Second Section Header: it doesn't matter how much text you put in a section header. The only thing
that matters is the text unit starts with an * and you use +consistent+
keywords for section headers if you want to use them for auto-coding.            23
                                                                                 24
Blah blah blah                                                                   25

*apostrophes and quotation marks                                                 26
                                                                                 27
Word automatically formats your apostrophes, in a way that CAN’T be saved        28
clearly in a text file. You may want to search for the curved                    29
apostrophe's and quotation marks and replace them with regular vertical          30
ones in K-Edit. It’s not a huge big deal, because you get used to the            31
weird characters, but it does make things somewhat easier to read in N5.         32
This is a quote: “The same goes for quotation marks.”                            33
So is this: "But here I use the regular straight vertical quotation              34
marks."                                                                          35

*One last word (section, really) on formatting                                   36
                                                                                 37
Note that there are no bold or italics or anything in this sample                38
document. You can use CAPS, +plus signs+, or +BOTH+ to indicate emphasis         39
instead, because N5 can't read formatting.                                       40
```

b. What goes in a document?

- An N5 document is usually just an MS Word file.
    - In this case, you would "import" a file, creating a new separate document.
- If you want to count "documents," you can define them in other ways.
    - Adding additional interviews with the same person to that person's document can be useful for longitudinal follow-up.
    - In this case, you would "append" additional text to an existing document.
- More on Importing Versus Appending
    - Importing means that you will have a new separate document in N5. Appending means that you will add text to an existing document.
    - If you intend to count teachers or caseworkers or something, you might want to have one document for each caseworker. That would mean that if you do an interview with Ms. Arroyo during the first round of site visits, you would *import* her interview into N5, but if you interview her again in the second round of visits, you would *append* the text of the second interview to the document containing the first interview. In any case, it will be crucial to be consistent about decisions like this.
    - If you decide to append text, you'll also need to be very careful to append the right text to the right document. In other words, if you're keeping multiple interviews with the same caseworker in the same N5 document, make sure that all the interviews in that document are with Ms. Arroyo and all of Ms. Arroyo's interviews end up in that N5 document.
    - This is another example of how getting the details right up front can make searches and fact-checking much simpler later on.

c. Text units

- A text unit is usually just a line of text (as in Chapter II), but a text unit can also be a paragraph or a sentence.
    - In Opening Doors, a text unit was a "mention." A new paragraph (a new text unit) starts when a different person speaks.

  - Because text units as sentences are a relatively new feature, we haven't had much experience with it.

- Below are some issues to consider in deciding how to define text units for your project:

  - *Smaller text units allow for more fine grained coding.* Text is filed by text unit; so, for example, an individual word cannot be coded if the text unit is a sentence, line, or paragraph.

```
1   So let's say that you have long text units. Now I'm going to say something
    about how the school just got a new cafeteria, and it's really great because
    the food is much better and it makes the kids happier, and I've even gotten
    food from there a bunch of times, and really my favorite is the macaroni and
    cheese, but I'm on weight watchers, and macaroni and cheese isn't good for
    my diet, but they've been talking about having a new salad bar in there, and
    Oh, right! The new curriculum, yeah, I've pretty much been ignoring it, be-
    cause I tried to use it and a one of the kids came in the second day with a
    complaint from her parent about it.
2   If you want to just code the text about the curriculum, you can't. You're
    going to have to read about the cafeteria every single time.
```

  - *Longer text units facilitate searching for quotes,* for example searching for quotes during fact-checking. The text searching capabilities look for the specified string of characters within one text unit, not across text units. So, if you were searching for "specified string of characters" and you had the text units below, you wouldn't find it.

```
1   So, if you were searching for "specified string of
2   characters" you wouldn't find it here, because it
3   is broken across two different text units. You
4   would have to do the search again, telling N5 to
5   look for "specified string" which would also pick
6   up "specified string of garbage."
```

- *The devil is in the details.* If you are going to count text units, then you need to be certain that they are consistently defined. The Opening Doors team defined a text unit as a "mention," one person making one comment in a focus group, so replicating their work, you would need to be certain that each "mention" really corresponds to a paragraph. For example, if a focus group participant gave a particularly long comment that was divided into separate paragraphs, you would need to combine the paragraphs into one so that they would be one text unit.

  - One way to check for this in Word, is under the "Edit" menu, choose "Find" and then type "^p" (without the quotes) for "paragraph mark" and just go through your document clicking "find next" and checking that each paragraph mark should really be there.

d. Document description (a.k.a. document description)

- A header can be printed in reports with excerpts from documents, to give the reader a reminder of the context from which the quote/excerpt is drawn. Also, when you click on the document in the document explorer, you can see the header. I have found that the "ideal header" differs from report to report, but basic substantive information like "site" can never hurt.

- Headers cannot be coded or searched though, so if you put information like site only in the header, you will not be able to auto-code the document for site. (The Jobs-Plus team resolved this issue by copying the header into each document twice with a blank line in between, so that the first copy was read in as the header, and the second copy was read in as part of the document itself, just a string of section headers with no text between.)

- If the first text unit of a document begins with an asterisk (*), then that and the following text units beginning with asterisks are read in as the document "header."

e. Sections

- A section is one or more consecutive text units in a document.

- A section begins with a text unit where the first character is an asterisk (*).

- Consistent section headers (section codes) are useful for automatically sorting text within documents.

- In the Jobs-Plus project, site researchers submitted reports that were divided into N5 sections that represent a number of different topics. In later rounds of the Urban Change Implementation study, they were used for sections of the interview guide, sets of questions and probes.

f. Keywords

- Remember that not all projects will use keyword coding. It's great if you can do keyword coding *while writing up your notes*. If you already have all of your documents and are ready to start coding, this is rarely the most effective method. (Instead, just bring your documents into N5 without keyword coding and see Chapter IV for other methods of coding.)

- First, make sure that the keywords are all spelled correctly. In order to take advantage of auto-coding, everyone needs to spell the keywords in exactly the same way. Using templates for written notes can help a lot in accomplishing this.

  – Remember the spell-checker poem (Appendix A).

  – In Urban Change, staff made small variations, and then research assistants had to code sections of text by hand.

  – Keywords do not necessarily have to have the same case, it's up to you to decide. When you write the program to do the auto-coding (see Chapter IV, for more information on how to do that), you can specify whether you want the search to be case-sensitive or not.

Methodological Innovations Initiative

Table 3

Keywords and Case Sensitivity

Search for:	**Actual Text in Document**	**Node that should be coded**	**Case Sensitive coding**	**Not Case-Sensitive coding**
#Site=Cleveland	#Site=cleveland	(1)	nothing	(1)
*Section1a	*Section1a	(2)	(2)	(2) and (3)
*Section1A	*Section1A	(3)	(3)	(2) and (3)

- Whenever possible, use document codes for basic information (like site or data type).

  – I would recommend that all projects and each and every document within the project use document codes to record basic information: site and date (e.g. month or quarter and year) at a bare minimum. This will help immensely with accessing documents down the road.

  – Document codes should appear at the top of the document and begin with a pound sign. So, for example, you could have #Site=Cleveland or just #Cleveland, at the top of the document. Multiple codes can be applied to the same document. So, you could have #Cleveland and #Line-staff. Or, you could even have document wide codes that refer to multiple sites, if the document is, for example, a memo about random assignment issues at more than one site. I recommend starting document wide keywords with the pound sign, so that when N5 searches for the text, it will have a unique string to look for. Otherwise, if you just searched for “Cleveland,” for example, and someone from LA said “We’re not like Cleveland, because…” the document would be incorrectly filed under Cleveland.

- Section codes may not always be useful.
  - Not every document will need section codes. For structured or semi-structured interview notes, it makes sense to use section headers in the way that you might use headers in your write up anyway. The same goes for documents from site researchers asked to address a particular set of issues, it might be really easy to include section headers for each issue, and if so that would make things much easier down the road. However, section headers don’t work as well for some forms of data, such as observational notes (which tend to be just a streaming narrative of what happened). Similarly, section headers can be more complicated to use with interview or focus group transcripts, because the conversation may jump around from one topic to another. While writing up our notes, we have the opportunity to organize notes by topic rather than chronologically, but that work is not included in the process of transcribing from a tape. In that case, one would have to read through the transcript several times in order to identify the relevant chunks of text and put them in the right category, and in that case, one might as well skip auto-coding and just do that within N5.
  - Section codes should appear at the beginning of the relevant section and begin with an asterisk. Note that formatting documents in this way does not allow for applying multiple codes to a particular section through auto-coding. The entire document might be coded to Cleveland and Line-staff, but the text under *Question-1 could only be auto-coded to Question 1 (in addition to Cleveland and Line-staff), it can’t also be auto-coded to *Question-2.

g. Word formatting

- N5 works with text only, but we don’t want to lose any meaning from the original document.
- Here are some suggestions for translating word formatting into a text only format. It’s best to use these formatting suggestions as you are writing up field notes, so that you don’t have to go through and make changes by hand later. When you bring the text into N5 you will automatically lose any formatting that N5 can’t understand.

- **Bold**, *italics*, font size, etc. I recommend using CAPITAL LETTERS and/or +plus signs+ instead to +EMPHASIZE+ text. (You could also use asterisks, like in e-mail, but then you just have to be careful about where they show up on the line, so that you don't unintentionally insert a section break.)

- Paragraph formatting

    - If you use indentation, line spacing, or formatted paragraph spacing to separate paragraphs in word, your paragraphs will probably look like they run together in N5. So, manually insert a tab in the first line of a paragraph and/or hit return an extra time in between paragraphs.

    - Bullets and other special characters will be saved as asterisks or something weird, and won't look the way they did in Word when they are in N5, so use something more simple like -, =, ~, or > instead. (Any character that actually appears on your keyboard will be fine.)

- *Headers and footers* won't show up in the text, so it's better not to use them. You might want to put what you would have put in a Word header or footer into the N5 header instead.

- *Tables and columns* are more difficult to translate into something that will make sense in a text only format, so if possible, it's best to avoid them.

h. *Anatomy of an N5 "project":* It's not just one file like in Word or SAS. Each N5 project has a parent directory named for the project, with two files (project.stp and n5.cfg) in that directory, and then several subdirectories:

- *Database:* This is where the actual project data is stored. There are tons of files in this directory containing the actual text, memos, and the coding structure and references to the text.

- *Commands:* This is where N5 will look when you tell it to run a command file. We'll talk about writing command files and saving them here.

- Raw files: This is where N5 looks first when you tell it to import new data, so you may want to store text files in the project in this directory.

- *Reports:* When you save output from N5, it will automatically be put in this directory.

- *Temp:* It's not clear to me what this folder does, but it will be there.

C. Starting up N5

1. When you first double-click on N5, a Startup box will appear. At the first screen you're asked to choose whether you want to open an existing project or start a new one.

   a. To open a project you've already started, choose 'Open an Existing Project.'

      - You may need to use the "browse" button, in order to find your project. If so, you'll want to find the actual project.stp file, in your project's directory.

   b. To create a new project, click on the button which says 'Start a New Project.'

      - A window will come up which is titled 'Select location of project.' Open the folders where you would like to save your project. Note that the name you are giving N5 is actually the name of the project's parent directory, not a file, like in Word. (N5 identifies projects by the name of the parent directory.) For example if you are creating the 'Urban Change' N5 project, when you type in "Urban Change," N5 will create a folder named 'Urban Change' in which to store your project.

      - Click 'Save' and another box will come up which says 'Please type your name'. Type your name there and click the 'OK' button. You will see the 'Document Explorer', 'Node Explorer' and 'Project Pad' windows.

2. Once you're working in a project, you'll see:

   a. Document Explorer, with a list of documents in the project;

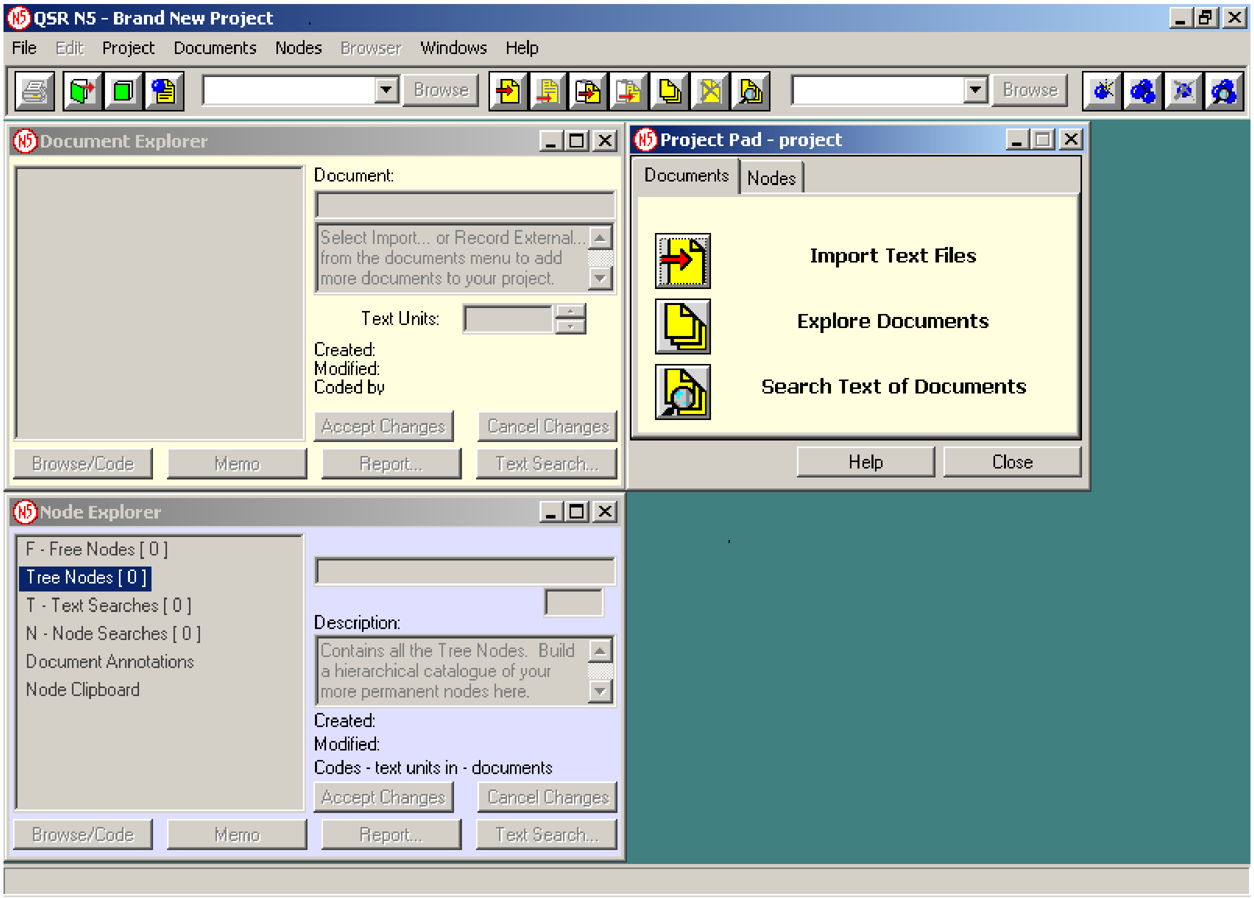
   b. Node Explorer, with a list of the nodes (or codes) in the project;

   c. Project Pad, which is N5's attempt to be more user friendly than the prior version.

Methodological Innovations Initiative

Figure 2

N5 Screen for a Brand New Project



Select Import... or Record External... from the documents menu to add more documents to your project.

Contains all the Tree Nodes. Build a hierarchical catalogue of your more permanent nodes here.

D. Reading Text into N5

1. First define your text units:

    a. Under the "Project" menu, choose "Preferences" > "Text Unit Type."

    b. A "Text Unit Type" window will pop up, and you can choose one of the options discussed above: line, sentence, or paragraph.

    c. The default for N5 is a line, and this is usually what we will use. But do check that this is set correctly either way, especially if you plan to count text units.

2. Using Copying and Pasting to Read One File into N5 at a Time

    a. Copy the document to the clipboard from Word

    - Open the document in Word
    - Under the "Edit" menu, choose "Select All" (or just hit Control-A)
    - Under the "Edit" menu, choose "Copy" (or just hit Control-C)

    b. To Import into N5

    - Under the "Documents" menu, choose "Import Clipboard Text as Document"
    - A couple of windows will come up telling you that N5 is working on it. They will be there longer for longer documents.
    - Rename the document:
        - In the top right column of the "Document Explorer" window, under "Online Document" it will say "Clipboard Text" or something like that. Click there, and type in the name you want the document to have. (I would recommend whatever the name of the word file is.) Then click on the "Accept Changes" button.
        - <u>This is really important!</u> If you forget to rename it, you'll have a bunch of documents called Clipboard Text, Clipboard Text 1, Clipboard Text 2, and so on, and it will be difficult to link them to the original word files, thus making it difficult to know what you've imported into N5 and what you haven't.

    c. To Append to an existing document in N5

- Click on the document you want to append to in the document explorer

- Under the “Documents” menu, choose “Append Clipboard Text to <name of the document you want to append to>”

3. Saving Word Files as Text to Read in Multiple Documents at Once

   a. Keeping track of original documents:

   - You can keep all of the documents in the “rawfiles” subdirectory of the N5 project, but if/when you merge projects from different team members, they won’t be copied into the new/merged file, so you might want to save them somewhere else on the network as well, in order to avoid sorting through old versions of the project. In any case, do save them all in the same directory. It’s just important to have copies of the original documents somewhere, just in case something goes wrong with N5, or in case you want to look at the original formatting for clarification.

   - Naming conventions can help keep track of which word files have been brought into N5. Use naming conventions to match documents in the N5 project to word files in these directories. You might want to include some basic information — e.g. author of field notes, date of interview, site — in the filename. As with file naming conventions for SAS data, the team should agree on and consistently use naming conventions. For example, in the Urban Change Institutional Project where each document was an interview with staff at one agency, we named a document:

     ```
     pken_rainbow
     ```

     The first letter ‘P’ stands for the site, Philadelphia, ‘ken’ is short for Kensington, and ‘Rainbow’ is an abbreviation of Project Rainbow, the agency name. The staff that worked on the Opening Doors project used names like:

     ```
     Macomb_A, Macomb_B, Macomb_C
     ```

     ‘Macomb’ was the site and the letter after the underscore stood for the type of focus group. ‘A’ is the current group of students, ‘B’ is the former group of students, and ‘C’ is the potential group of students.

   b. Step 1 in Word: Save the Word file(s) as a text file.

   - Using the Word macro:

- There is a Word macro that can convert all Word files in a directory to text files, this is *very* handy if you have a lot of files to import at once.

- How to install the macro (you only have to do this once):

  - Find strip2000.dot, on the MDRC intranet web page.

  - In Word, go to the "Tools" menu and choose "Options," then click on the tab labeled "File Locations."

  - A window will come up listing a number of directories on your computer, write down the location for start-up files (the directory listed next to the word start-up). If you can't see the full directory path, click on the "Modify" button, and just look at it in the window that pops up (press "Cancel" once you've written it down). For example, on my computer, the word start-up files are located in:

```
C:\WINNT\Profiles\widom.000\Application Data\Microsoft\Word\STARTUP\
```

  - Exit Microsoft Word, and copy the file strip2000.dot into your word startup directory, then exit Word and start Word again, and the strip macro will be available to you.

- Under the "Tools" menu, choose "Macro" and then "Macros," or type Alt-F8. A window will appear with several options for macros you can choose from, double-click on srcSTRIP.MAIN (or click on it once and click "Run").

  - Text Save Format: Always choose "text-only (text unit = paragraph)." N5 will put in additional line breaks as necessary, based on the preferences you specified above.

  - File Selection: You can either do one file at a time (in which case you would choose "current file" only), or you can process a whole directory of files at once (which is handy, choose "All files in source path").

  - Unprotect Documents (Batch Mode): Just leave this as checked "no." It deals with weird windows permissions issues. If you get an error saying that it can't access some of the files in the directory, then you can click "yes" and try it again, but most of the time this will be unnecessary, and if you check "yes" it will only slow the macro down.

- Source: This is where the macro will look for the files it's supposed to convert. If you chose "current file" then don't type anything here. If you chose "All files in source path," then enter the directory where the files are here. Note that the macro will process *all* files in that directory, so if there are files that you don't want to have processed, it's better to make another subdirectory for only the files you want to process. (You can always move them later.) For example, you might enter

```
Q:\MDRC project directory\N5-project-name\rawfiles\
```

- Output: Same as above, but this is where the finished text files will be put.

- File Specification: Just leave this as the default *.doc, because you can only convert MS Word files.

- Optional Prefix: If you want to change the file name from the original word document to the converted text document, you can use this feature. For example, if you are converting a bunch of files from Cleveland, you might want them all to have the site as a prefix, so that `adminint_08-01.doc` would be converted to `clev_adminint_08-01.txt`. If that were the case, you would enter "clev_" here.

- Optional Suffix: Same idea as above, but the converted file would be `adminint_08-01clev_.txt` if you types "clev_" here. Both of these are optional. You can leave them blank.

- Do not change spaces in text filename: Leave this box unchecked.

– After you click "OK" a window will appear saying "RTU Strip1.06 for NUD•IST will attempt to convert files" and you will see another message when the conversion has been completed.

- To save a document from Word without the strip2000 macro:

  – Open the file in word.

  – On the menu choose File > Save As

  – A window will open, and at the very bottom there will be a pull-down box labeled "Save as type." The box will probably say "Word document (*.doc)" at first. Choose “Text Only (*.txt).” This will allow N5 to read the documents, and insert text unit breaks as necessary, based on your preference specified above.

– Choose the appropriate folder and enter a file name as you normally would, and click the “Save” button.

– A warning box will probably come up that says that your file “may contain features that are not compatible with Text Only format. Do you want to save the document in this format?” *Click “Yes.”*

c. Step 2 in N5: Import the documents.

- Click “Import Documents” from the Project Pad. (Or, from the “Documents” menu, choose “Import Text Files as Documents.”)
- A window titled “Select files to import” will open, find the directory where you saved the files you want to import, click on one and then hold down the shift key and click on others in order to select them, and click the “OK” button.
- A window will pop up telling you that N5 is working on it, and then you should see the new documents in the document explorer window. N5 will automatically assign the windows filename as the N5 document name.

E. Where Are We Now? Document Preparation in the Sample Projects

1. Where are we now?

a. In Chapter II we talked about coding being the central activity of analysis, different approaches to doing the analysis.

b. In this chapter, we got started by bringing the data (text) into N5.

c. The next chapter covers the nuts and bolts of getting that analytic work incorporated in N5. How do we flag text and file in N5?

IV. Telling N5 What You Think: The Mechanics of Coding

A. Overview

1. Potential questions we could code for and setting up a node structure based on those decisions

2. Different methods for telling N5 about those coding decisions

    a. Coding by hand, when you want to read the text on your computer screen and code as you go.

    b. Using the "Quick Coder," when you have coding notes written in the margins of printed reports from N5. (This means you know the document names and text unit numbers already.)

    c. Using "auto-coding" when you have keyword coded documents that you want to file automatically (file basic information about the document, and topic coding done during document preparation).

    d. Importing tables from Excel when you have a list of basic characteristics of each of your N5 documents in Excel that you want to bring into N5.

B. Deciding What to Code

1. Remember from Chapter II that it's not a good idea to over-code. Keep it simple.

2. The coding you do should be guided by the types of questions the research team wants you to answer. For example see Table 4:

Methodological Innovations Initiative

Table 4

Potential Research Questions from Sample Projects

Instruction	**Urban Change**	**Jobs-Plus**	**Opening Doors**
List text on a particular topic.	Show me all of the responses to the caseworker question about enforcing participation requirements (sanctioning).	Show me everything about resident participation, from templates and other materials.	Show me everywhere that students talked about social supports for attending school.
List a subset of the text on a particular topic.	Tell me what the caseworkers in Cleveland said to the question about enforcing participation requirements (sanctioning).	Show me everything on resident participation (from templates and other materials) for Cleveland.	Show me what current students said about social supports.
List information on a particular topic in sections based on a set of categories.	Give me the answers to the question about enforcing participation requirements (sanctioning) by site.	Give me the information on resident participation (from templates and other materials) by site.	Show me the information on social supports by type of student.
Create a cross-tab table.	How many staff at each level did we interview at each site?		Show me the number of focus groups of each type from each site.
Count based on coded themes.	How many subjects said that they are responsible for sanctioning and consider it to be an important part of their jobs?	How many sites had dropped collaborative members? (See CHComp4, in Section A)	How many focus groups talked about child care as an important social support?
Find people who said one thing in one part of an interview and another in another part.		Show me the sites that mentioned dissatisfaction with a collaborative member *and* had never dropped any collaborative members.	

3. Below are a list of codes that you would need to implement in order to answer *all* of these questions. *But*, keep in mind when planning your own team's work, that you don't need to code everything at once, and it's always better to start with a smaller more general list, and then move towards more specific coding in successive rounds.

    a. Urban Change

    - Basic coding: Site (Cleveland or Philadelphia) and Staff Level (Caseworker, Supervisor, or Administrator)

- Topic coding: Based on interpreting the subheadings, in particular looking for questions on sanctioning.

- Coding for themes: Looking for themes within the text on sanctioning, in particular being responsible for it and thinking of it as an important part of their work. (Could code this in two separate nodes and search for people who said both, or just code those who said both; the most important thing is to think how it will be used and to be consistent within the team.)

b. Opening Doors

- Basic coding: Site (Macomb or Valencia) and Focus Group Type (Current, Former, or Potential students)

- Topic coding: Social (Non-financial) supports for attending school.

- Coding for themes: Looking more closely at the text on social supports, create a code for references to child care in particular.

c. Jobs-Plus

- Basic coding: Site (Baltimore, Dayton, or Cleveland) and keyword coded topic areas for the template documents (see keyword list, sub-topics within composition, governance, technical assistance, agencies, residents, and other observations).

- Topic coding: Information on resident participation (from non-template documents); and information on dissatisfaction with collaborative members (listed under sub-heading "SatComp2" under composition in the template) and on dropping collaborative members (listed under sub-heading "ChComp 4" in the section on composition in the template).

- Coding for themes: Looking more closely at the text on dissatisfaction with a collaborative member and dropping collaborative members, code to sort the answers into groups (i.e. those sites that had dropped collaborative members and those that hadn't).

C. Command Files

1. Throughout this manual there are instructions for performing tasks using menus and using command files. Here are the basics of what a command file is, and how to use them.

2. Choose "File > New" from the menu. Note that this is a new file, like a word document, this is not the same as creating a new N5 project.

3. In the window that opens, type in your commands. (More information on specific commands run throughout the manual.) It's always a good idea to start with a general description of what the command file does, who wrote it, and when. Be sure to use comment lines (that begin with a semicolon) for this description. For example,

```
;;; this is a sample header for a command file in N5
;;; written by Rebecca Widom, on 5-24-02
;;; this program doesn't do anything, because these are just
;;; comment lines.
```

4. Once you have entered the commands — to create nodes, do searches, whatever — save the command file somewhere that you can easily remember, possibly in the commands subdirectory of your N5 project, though, you'll have to re-copy it in by hand if you merge projects later.

5. To edit an existing command file later, just choose "File > Open" and edit away.

6. To run a command file, select "Run Command File" from the "Project" menu, and select the file that you just saved above.

7. Note that you can't "undo" a command file, so before you run one, it's always good to make sure that you have a backup copy of the project available in case something goes wrong.

D. Creating a Node Structure

1. Before you can start coding, you have to have "nodes" that represent each of your codes. This step is analogous to labeling a bunch of file folders, before you actually put anything in them.

2. So, we'll start with creating a node structure.

   a. Urban Change

```
(1) Site [Note that nothing gets coded to this node.]
(1 1) Cleveland
(1 2) Philadelphia
(2) Staff level [Note that nothing gets coded to this node.]
(2 1) Caseworker
(2 2) Supervisor
(2 3) Administrator
(3) Sanctioning
```

(3 1) Responsible-for
(3 2) Important

b. Opening Doors

(1) Site *[Note that nothing gets coded to this node.]*
(1 1) Macomb
(1 2) Valencia
(2) Group-type *[Note that nothing gets coded to this node.]*
(2 1) A-Current
(2 2) B-Former
(2 3) C-Potential
(3) Social supports
(3 1) Child care

c. Jobs-Plus

(1) Site *[Note that nothing gets coded to this node.]*
(1 1) Baltimore
(1 2) Cleveland
(1 3) Dayton
(2) Template-topics
(2 1) Comp
(2 1 1) SatComp2
(2 1 2) ChComp4
(2 2) Gov
(2 3) TAMDRC
(2 4) TATEN
(2 5) TACOI
(2 6) TAOther
(2 7) PHA
(2 8) Wel
(2 9) JTPA
(2 10) OtherPart
(2 11) Integrate
(2 12) PHARes
(2 13) RACRes
(2 14) BroadRes
(2 15) JPRACRes
(2 16) PowerRes
(2 17) FactorRes
(2 18) OtherObs
(3) Non-template topics
(3 1) Residents
(4) Themes
(4 1) Dropped members
(4 1 1) Yes
(4 1 2) No
(4 2) Dissatisfied w/members
(4 2 1) Yes
(4 2 2) No

3. Types of nodes: Above, I've called everything a tree node, but you don't have to do it that way. Don't worry about differences among text search, index search, free nodes, and tree nodes, except for three things:

    a. If you will want to create tables or reports by site, for example, you will need to use a hierarchical node structure. (I sometimes call this a "vector node," because they are useful for vector searches.) For categorical variables you plan to use in tables, create a "parent" node (e.g. Site) *with no text coded to it*, and "child" nodes (e.g. Cleveland, Philadelphia, Miami and LA) which have text coded to them. <u>All of these have to be tree nodes.</u>

    b. When you do an index search, N5 will automatically store the results (the text you told it to find) in a new node under "index searches." Similarly, a text search will result in a new node under "text searches." Beyond the difference in how they are created, though, you can treat these nodes as you would any other. (More on searching in Chapter V.)

    c. When you use the build-tree macro command, it will automatically create tree nodes.

4. There is more than one way to create non-search nodes.

    a. By hand:

        ▪ Decide where you want the node to appear — is it a free node, a top level tree node, or a new "branch" off of an existing index tree node?

        ▪ In the "Node Explorer" window, select the parent of the node you want to create — "Free Nodes," "Tree Nodes," or an existing node in the index tree.

        ▪ Then, right click on the node and choose "Make free node," "Make top-level Tree Node," or "Make new Tree Node" from the menu that appears. (The menu will say different things as N5's way of confirming where you clicked in the step above. Do you really want that to be a free node? Etc.)

        ▪ Enter the node name, and a description if you want, in the node explorer window and click "Accept Changes."

        ▪ You can also give the node a particular number if you want, for example, to space out the base nodes at 100 and 200 or whatever.

    b. By command file:

- Creating nodes in a command file can be very useful, if you have a bunch of nodes that you know you want to create all at once, such as a list of keywords that you want to make into nodes for auto-coding. (Note that it's best to put this in a separate command file from the auto-coding command file, because otherwise, every time that you want to auto-code some documents, you will create a new additional set of identical nodes that will need to be deleted.)

- Use the "build-tree" command.

  – Each node name needs to be enclosed in quotation marks, and lists of "child" nodes are enclosed in parentheses.

  – As an example, the code below will build the tree described above for the Urban Change project.

```
(build-tree
    ("Site"
       ("Cleveland"
       "Philadelphia")
    "Staff level"
       ("Caseworker"
       "Supervisor"
       "Administrator")
    "Interview Section"
       ("Responsible-for"
       "Important")
    )
    )
```

- N5 will automatically assign node numbers, but you can choose where to build the tree from, by adding a couple of lines before the last parenthesis. For example:

```
 (build-tree
    ("Site"
       ("Cleveland"
       "Philadelphia")
    "Staff level"
       ("Caseworker"
       "Supervisor"
       "Administrator")
    "Interview Section"
       ("Responsible-for"
       "Important")
    )
    node (1)
    node-title "Build-tree command nodes"
    )
```

- Note that sometimes with the build-tree command, you get an error even though the program worked fine. I don't know why, it's a quirk with N5.

5. The instructions below assume that you have already created the nodes you will need. Keep in mind that you can always create new nodes as you need them.

E. Applying Coding

1. The analogy: Now that you have labeled your file folders, this is how you tell N5 which text to put in each one. There are myriad ways to enter coding, these are only the ones that have seemed most useful at MDRC.

2. The Basics: Reading text on your screen and coding as you go

   a. Use this when you want to read the text on your computer screen and code as you go. This method works equally well for a first pass through a document and for further coding of text on a particular topic.

      - The disadvantage of working this way is that there is no record of your coding decisions (why you interpreted particular text in a particular way), and no easy way to revisit those decisions later.

      - One option is to do an initial pass through of coding in this way, filing questionable text in a particular node, so that you can document and revisit those specific decisions later. This strategy works best when there is only one person working with the data, though. On a team, it creates a version control and communication nightmare.

   b. This is *not* a good method when you want to do basic coding that applies to the entire document, because it is a pain to select all of the text in the document. In this case, it's better to use the quick coder or import tables from Excel, described below.

   c. There are two types of browsers that you can use to read text on your computer screen:

      - Use the document browser if you want to read through entire documents. This would generally be best for topic coding, earlier in the analytic process.

      - Use the node browser to code text on a particular topic. For example, you might want to code responses to a particular question in an interview guide.

d. Open the *document browser or node browser window* by clicking on the relevant document or node and hitting the “browse” button in the document or node explorer window.

e. Click on the node you want to code to in the node explorer or select it from the available “recently used” list at the bottom of the browser.

f. Select the relevant text in the browser window.

g. Click the “Code” button at the bottom of the browser window.

3. Using a coding template in Excel

   a. This is a strategy that allows coders who don’t want to or can’t work directly in N5 to code documents without creating a lot of extra data entry work for the data manager. This also allows teams to check coding decisions for typos more easily before they are entered.

   b. Creating the coding template takes some care. There are sample coding templates available on the MDRC intranet page, that you can adapt for your project. Or follow these directions:

   - Set up the first row as follows:

	Column A	Column B
Row 1	Document name:	[document name, exactly as it appears on the N5 report, but enclosed in quotes]

   - Below that, enter the following column headings. Then make a list in columns A and B of all of the node addresses and corresponding node titles in your project. Check that these match your project exactly, or else the template won’t work.

	Column A	Column B	Column C	Column D
Row 2	Node address	Node Title	Text units to code	;;N5 command

   - Enter the Excel formulae that will generate N5 coding commands:

     – For rows with basic document codes (e.g. row 7 here), column D would contain this formula: =`IF(C7="x",CONCATENATE("(add-coding ",$B$1, " ",A7,")"),"")`

        - For rows with topic codes (e.g. row 36 here), column D would contain this formula: `=IF(ISBLANK(C36),"",CONCATENATE("(add-coding ",$B$1," ",A36," text-units (", C36,"))"))`

    - Once you've entered the formula once, you can copy it to all of the other document rows or topic rows accordingly.

c. Once the template is set up, you can send it to team members with printed document reports (see Chapter V) and they can enter the coding into Excel as follows.

    - At the top of the template (in cell B1) enter the document name, exactly as it appears on the N5 report, but enclosed in quotes, e.g. "BcollintHABC041699".

    - Check off the basic/whole document codes that apply to the document by entering an x next to them in column C. (When you do that, an N5 command should appear in column D next to them.)

    - For the topic codes, enter specific text units next to them in column C:

        - To code one text unit, just enter that number: 365

        - To code a range of text units, enter just the first and last unit enclosed in parentheses: (325 335)

        - To make a more complicated list, use the rules above and separate list items with a space: (325 335) 365 (1 10) (24 36)

    - Copy the command file from Excel to N5:

        - First, in Excel copy Column D.

        - Then, back in N5, choose "File > New" from the menu, and paste the command file into the N5 text editor by typing Control-V. Save the command file in your N5 project's command file subdirectory.

d. Copy the command file from Excel to N5:

    - First, in Excel copy Column D.

    - Then, back in N5, choose "File > New" from the menu, and paste the command file into the N5 text editor by typing Control-V. Save the command file in your N5 project's command file subdirectory.

4. Using “auto-coding”

   a. Use this when you have keyword coded documents that you want to file automatically (file basic information about the document, and topic coding done during document preparation).

   b. Auto-coding is basically automatic filing. Once you have written the command file, it allows you to automatically translate keyword coding in documents into coding in your N5 project.

   c. Make sure that you have a node for each keyword (or create one), and check to make sure that the keywords are inserted correctly in your documents.

   d. Create the command file (note that you only have to do this once).

      - Create a spreadsheet in Excel, with the following columns: This is the very most basic format, but you can modify it.

        – I find it helpful to have column headings in the top row, so the formulae below refer to information starting in the second row of the spreadsheet.

        – Column A: Keywords. A list of the keywords as they appear in the documents, with either an asterisk or a pound sign.

        – Column B: Node address. Type in the address for the keyword’s node.

        – Column C: Spread. Specify whether this keyword applies to the document, section, or text unit. If you’re using keywords starting with * for section codes and starting with # for document codes, you can use the Excel formula below to fill in this column. Note that if a cell says “Problem!” you’ll need to check that your keyword is entered correctly, and that there are no extra spaces at the beginning of the text in the cell.

```
=IF(LEFT(A2)="*",    "to-section",    IF(LEFT(A2)="#","to-document",
"PROBLEM!"))
```

        – Column D: Macro Command. Here you would enter the following concatenation formula in the first row under the column headings and then copy it down the column to generate additional commands.

```
=CONCATENATE('(search-text  "',  A2,  '"  whole-word?  yes  case-
sensitive? no node ', B2, ' spread ', C2'))')
```

- Copy the command file from Excel to N5 following instructions above.

e. If you are planning to run the program on all of the documents currently in the N5 project, then you're all set. You might want to do that if, for example, you're starting a new project and want to auto-code all of the documents at once. But if, for example, you're getting interviews from a round of field research one site at a time, each time that you bring new interviews into the project, you would just want to code the new interviews, and not spend time re-coding a bunch of files. It wouldn't hurt to re-run the auto-code program on those interviews, it would just be slower.

- To run the auto-code macro on a few documents only, first create a note called "to auto-code," for example, and code the new interview documents — the whole document, not just certain sections — to that node. Then, use the "set restriction" command at the beginning of your macro and the "end-restriction" command at the end to run your macro just on those documents. So your command file might look more like this. (Text <in brackets> is information that you'd need to fill in.)

```
;;; Filename: <filename>
;;; Created by <your name>, on <date>
;;; <brief description of what this command file does.>

(set-restriction only text <address for "to auto-code", would look
like "(200 1)">)

<search text commands, copied from Column D in Excel>

(end-restriction)
```

- You can always go back and edit a command file later, so you can run it first without the restriction and then put in the restriction in the future if you need it.

5. Importing tables from Excel

a. I suggest using document codes within the actual documents themselves for auto-coding basic information, because there are fewer files to keep track of that way, and you have more flexibility with which documents you actually code.

- Note that this strategy is designed for mutually exclusive categories, keyword coding is much more flexible in this way. (So, for example, using an import table strategy for the Jobs-Plus documents, under site, we would need to create a category called "Multi." You can't have a document coded to multiple sites.)

b. Use this when you have a list of basic characteristics of each of your N5 documents in Excel that you want to bring into N5.

    - Some projects include document information in the file name, and importing codes from Excel can be particularly helpful in those cases.

    - If you have data in SAS that you want to link with your documents (for example, if you had done in-depth interviews with a subset of survey respondents), importing data from Excel can be useful for that.

    - On the Urban Change Institutional Study, we put a fair amount of effort into coding the type of services offered by each agency, and documented these coding decisions in Excel, and then ultimately applied the codes to the entire interview document.

c. Set up the spreadsheet

    - Create a list of the document names in column A of your spreadsheet. In the first row of column A, use the title "Document," so that N5 knows that this table refers to documents as opposed to cases.

        - Note that the names all have to be spelled exactly the way that they appear in the N5 document browser window.

        - If you have a lot of documents, you may want to use the "Documents Report" feature in N5.

            - Go to the Documents menu in N5 and choose "Documents Report." When the window comes up, make sure that none of the check boxes are checked and that next to the "Restrict" button, it says "No Restriction," then click "OK." You'll get a report with the number and title of each document in the project, listed in paragraph form.

            - Click the X to close the report window, and a dialog box will come up asking if you want to keep the report. Click, "Yes," and save the report somewhere that you can easily find it. (You'll want to pay attention to the directory here, because N5 will put it somewhere weird on your C drive if you don't.)

            - Open the file in Word or KEDIT and delete the document numbers and any extra spaces, so that you are left with a list with one document name on each line.

- Copy all of the document names from Word or KEDIT into Excel.

- Create columns for the "parent" codes you want to create — for example, "Site" is a column, but "Cleveland" would be an entry in the column, not it's own column. You can have as many or as few columns in one spreadsheet as you want.

  - Put the name of the parent node (e.g. Site) at the top of the column, and fill in the name of the child node (e.g. "Cleveland") for each document on the list.

  - Note that if you put "Cleveland and Philadelphia" in one of the cells, N5 will create a separate node called "Cleveland and Philadelphia," rather than coding the document to both the Philadelphia node and the Cleveland node.

  - Similarly, if you put "Philadelphia" in one row, "Philly" in another, and "Phildelphia" in a third, you'll end up with three separate nodes with those titles.

  - You can use Excel formulae (the "if" function with text functions like "right," "left," and "substr") to read codes from the file names, or just type them in by hand. For example, with Opening Doors, where the last letter in the filename indicates the type of focus group, you could use the formula below. Note that if the filename does not start with A, B, or C, the formula will return an error.

```
=if(right(a2)="A",   "A-Current",   if(right(a2)="B",   "B-Former",
if(right(a2)="C", "C-Potential","ERROR!")))
```

- When you've finished, save the excel file as a text file, in the rawfiles subdirectory of your N5 project. In Excel, choose "File," "Save as," and then in the window that comes up, where it says "Save as type" choose "Text (tab delimited) (*.txt)." A box will come up that says something like "file may contain features that are not compatible with the format you chose, are you sure you want to do this?"; click "Yes."

- Once the file is saved close it and/or exit Excel, otherwise N5 will not be able to access the file.

- In N5, you will need to first set up a parent node — for example, called "Base Data" — for the nodes in your spreadsheet. When you import the spreadsheet, N5 will automatically create a structure for each column of the spreadsheet, under that parent node.

- Once you have created the parent node, go to the "Project" menu and choose "Import table." Then double click on the file you just saved: the text version of the spreadsheet from Excel. Then, a window will ask for a node, and you should select the parent node that you created above (you can either type in the address or use the "select" button to find the node). Finally, N5 will confirm that you want to make these changes, say "yes."

- Note that if N5 asks for a node with cases, you have probably misspelled one of the document names in your spreadsheet or forgotten to use the column header "Document." You can also code cases rather than documents in this way, for example, if you wanted to code individual focus group participants in Opening Doors, where the document is actually the focus group itself. (This manual doesn't cover working with cases, because they are not well integrated into the reports function. For more information on cases, see pages 118-122 of the *N5 Reference Guide*.

- Note that if you import additional documents later and want to code to these nodes, you will have to either a) code them by hand, or b) delete the nodes, add the new documents to the previous spreadsheet, and re-import the entire new spreadsheet.

F. Spreading Coding

1. When you might want to "spread coding":

   a. You're coding basic information in the browser and you just want to select a little text but then apply the coding to the entire document.

   b. You have a node that is coded too narrowly, either because it comes from a text search or because of miscommunication with the person who did the coding, and you want to get a little more context around the quotes that you're reading.

2. <u>This is not the same as re-evaluating the coding of a particular passage of text.</u> When you spread coding, you spread around each and every passage of text coded to that node.

3. <u>How to spread coding:</u>

   a. Right Click on the node you want to spread.

   b. Select "Spread all coding" from the menu that pops up

   c. Choose how much text you'd like to include. When in doubt, it's better to be conservative here, and err on the side of spreading less, because you can always spread more later. You can't spread less though.

- Certain number of text units on either side. The default is 5, and this makes sense if you have one line of text as a text unit, and you just want a little more context.

- Paragraph. This is all of the text units before a blank one. If your documents are formatted consistently into paragraphs, this would probably be useful for getting a little more context.

- Section. Same idea as the paragraph, see Chapter III for more information on defining sections.

- Document. This is useful if you are applying a basic code to the entire document. It's not a good idea if you just want some more context.

V. The Fun Part: Creating Output/Reports

A. Overview

1. The big caveat: This is what we've been waiting for! This is how you get fabulous information back from N5! Just remember that the information will be only as fabulous as the work you've done up to this point.

2. The basics, node reports and document reports: what's the difference, how do you create them, practice creating basic reports.

3. Using searches to create tables and fancier reports: what do the search operators mean, when would you want to use them, practice creating some fancier reports.

4. Tips for fact-checking.

B. The Big Caveat

1. This is the really fun part. Once you've set up your documents and done the coding, you can get really cool reports really quickly to answer questions that would take a ton of time using highlighters and tape flags. This is where you get the pay-off from using N5.

2. **But, any report that you generate is only as useful and accurate as the work you've done up to this point.**

   a. You can print a report of all the information on sanctioning in about two minutes, but if the person who coded the information on sanctioning missed a bunch of stuff or included a bunch of stuff that was irrelevant, the report you'll get won't be very helpful.

   b. Similarly, we'll show how N5 facilitates fact-checking. However, if you haven't imported all of your data into N5, done the coding in N5, explained how the documents and codes relate to what you've written, simply checking on a node won't verify that "All [17] of the Philadelphia case managers also recommended English as a Second Language training for a limited-English speaking recipient." (UC Implementation, Chapter 4 and Table A.1)

C. The Basics: Creating, Saving, and Printing Reports

1. What's the difference? Document versus node reports:

    a. Documents are the basic materials that you bring into N5. So, in Opening Doors, a document is a focus group transcript. In Jobs-Plus, a document could be an annual site report, a monthly memo, an e-mail, whatever. In Urban Change, a document is a write up of an interview or an observation. Especially when you're just starting the analysis, you may want to print out a set of documents just to get the whole picture of your data.

    b. Nodes are like file folders or different colors of highlighters or tape flags, and they represent a topic or a theme or a basic sub-set of the data like a site. Further on in the analysis, you may want to print out node reports to take a closer look at the data on one particular topic, such as what the caseworkers said in response to our question about sanctioning, in order to look for themes. A special kind of node, called a matrix or vector node (which we'll talk about below) can let you to look at a topic separated by another set of categories, such as, the information on sanctioning by site.

    c. Coding in N5 links nodes and document text. It's like putting documents in folders or marking specific text with different colored highlighters.

2. This chapter focuses on creating printed output from N5, but you can also browse documents or nodes live from the screen. (Just click on the node or the document and then click the browse button in the node or document explorer window as appropriate.)

3. Creating a document report:

    a. Click on the document in the document explorer window.

    b. Click on Report.

        - Choose the options that you want:

    c. Text to display:

        - Document description: see below (under creating a node report) under "with headers" it's the same thing.

        - Document memo: You can also write a memo about each document. I tend not to use memos and don't discuss them here, so I wouldn't check this.

- Document text: You almost always want this, it's the actual text in the document. The only time that you wouldn't check this is if you were doing an informal inter-rater reliability check. For more information see "Summary only" coding below.

d. Coding to display:

- No coding: This is pretty self-explanatory. Check this if you don't want to see any coding information.

- Summary only: This is what you would use to get a list of all of the codes applied anywhere in the document and which text units (usually line numbers) they apply to. It's useful if you have everyone code the same document, and you want to check that you all coded it the same way. You can produce this list with or without the actual text of the document included in the report. See "document text" above.

- Coding stripes: see description below.

- Cross-references: see description below.

4. Creating a node report

a. Click on the node in the node explorer window.

b. Click the "Report" button in the node explorer window, and choose which type of report you want:

- "Show Node Data" is information about the node (the number of documents coded to it, the memo if there is one, the description if there is one, etc.) but not the text actually coded to the node. You will probably not use this option much.

- "Show Coding Data" is about the actual text that is coded to the node. Choose this.

c. Below "Show Coding Data" there are several more options:

- *Document descriptions:* If you have used document descriptions (a.k.a. document descriptions), you can have them printed out with the text from the document. This might help to contextualize the text that you're reading.

- *Coded text:* Choose this, because if you don't you will only get a list of document names and text units.

- *With Sub-headers:* Rarely too helpful. If you have, for example, divided your document into sections based on an interview guide, and then coded for a theme that might show up in multiple sections of the interview, you might want to know, when looking at each text segment, when it was mentioned in the course of the interview. If this is your situation, check here. Otherwise, don't bother, if you're printing codes based on auto-coded sections, you'll get the section headers either way, because they are coded already.

- *With coding stripes:* This is another way to get some basic information about the text that you're reading. So, for example in Urban Change, you might want to look at a topic by site, but then also know if the interview you're reading was with a caseworker or an administrator. So, you could choose coding stripes, and when you say "ok" N5 will pop up another window asking which nodes you want stripes for, and then it will put letters in the margins to indicate which text is coded to those nodes. This can be helpful if you have a simple thing you want to look at (e.g. caseworker or administrator, A or B in the margin), but if you choose too many nodes, you'll have a bunch of letters in the margin and you'll need to go back to the beginning of the report to remember what they mean.

- *One Document Only:* This is pretty self-explanatory. If you choose this, N5 will ask which document you want to see. Generally speaking if you're supposed to be printing all of the information on a particular topic, don't choose this.

d. Cross-References: This prints out, under each text unit, every other node that the text unit is filed under. It's generally not too helpful. If you choose to do it though, you can select "With node titles" and then N5 will print out the node title (e.g. Cleveland) rather than just the numeric node address (e.g. "(100 2)"). This is more useful if you have longer text units. If every line is a text unit, then it can just be annoying to read the same list of numbers over and over.

5. Saving reports as text to use in Word or KEDIT

a. By default, reports are saved in the "reports" subdirectory of your project.

b. Saving files is pretty straight forward, either choose "File" "Save" from the menu, or when you go to close the window it will ask if you want to save, or it may prompt you to save the text if the entire report is too long for one N5 window.

c. Then you can open the report file in Word or KEDIT. Using 4Print to print from KEDIT, automatically formats the text to save a lot of paper over printing from Word.

You should have KEDIT already installed on your computer, it's a pretty standard MDRC application, but if not, contact the help desk or just use Word.

d. To use 4Print in KEDIT, click on the button at the bottom of the screen that says "4P". A box will open asking you for optional commands. It's probably best to just click "ok." The default (probably something like 20 -2 -D) is probably what you want (in this case the 20th floor printer, doubled sided printing, with two columns per page). Just in case, here is an explanation of the different options.

    - The printer (or what floor you're on): 18, 19, or 20

    - The number of columns of text per page (using 2 columns still leaves the text legible and saves many trees): -1 or -2

    - Whether or not you want the printing to be double sided. If so, use: -D

D. Searching

1. Now we've covered the most basic types of reports — give me everything coded to a particular node or give me this document. But, N5 also allows you to build on the coding that you've done to get more specific sections of text, by doing Node searches.

2. What kinds of searches would you like to do? Some possible examples:

    a. What did caseworkers say about enforcing participation requirements (sanctioning)?

    b. What did current students say about social supports?

    c. Show me the information on social supports by type of student.

    d. How many staff at each level did we interview at each site?

    e. How many sites mentioned dissatisfaction with a collaborative member and had never dropped any collaborative members?

3. Table 5 describes some common types of reports and kinds of searches that you may need to do to create them.

    a. The first column is the type of question, with some specific examples from our sample projects.

    b. For each question in the first column, the second column has one or two ways that the project might be set up that would allow you to answer the question.

Methodological Innovations Initiative

Table 5

Sample Search Questions

Instruction	What you have coded, advance work done	Searching Steps
List text on a particular topic. ▪ What did caseworkers say about enforcing participation requirements (sanctioning)? ▪ Show me everything about resident participation, from templates and other materials. ▪ What did students say about social supports for attending school?	One node containing all of the relevant text (e.g. everything on social supports)	*None necessary*, just make a report for that node.
	A number of nodes which each individually cover some piece of the information and together encompass all of the relevant text (e.g. specific nodes on resident information from the templates and one separate node on resident information from the non-template materials).	*Union search*: select all of the relevant nodes, in order to create one node containing all of the questions.
List a subset of the text on a particular topic. ▪ What did caseworkers in Cleveland say about enforcing participation requirements (sanctioning)? ▪ Show me everything on resident participation (from templates and other materials) for Cleveland. ▪ What did current students say about social supports?	One node for sanctioning or Section A questions (or see above about creating one) and another node containing all text from interviews with caseworkers and/or information from Cleveland.	*Intersect search*: select the Section A node and the caseworker node and the Cleveland node.
List information on a particular topic in sections based on a set of categories. ▪ Give me the answers to the question about enforcing participation requirements (sanctioning) by site. ▪ Give me the information on resident participation (from templates and other materials) by site. ▪ Show me the information on social supports by type of student.	▪ The node for sanctioning, resident participation, social supports, etc. ▪ A vector node for the categories you're interested in (parent node called site with nothing coded to it, and child nodes for the actual sites with all of the text from each site coded to the appropriate site node). This means that all of the Section A text units are also coded to a site node.	*Vector search:* Select the Section A node (for the left box) and the site parent node (for the right box). Choose *Intersect* from the pull down menu in the middle. And click ok. When you create a report from the vector search, it will list the Section A text by site.
	Your site codes only apply to part of the document, but not the entire document, so some or all of the relevant text may not be coded to site.	*Vector search*: Select the Section A node (left box) and the site parent node (right box). Choose *Near* from the pull down menu in the middle. Click "context" and "document." Click "retrieve" and "first" to indicate the Section A node.

Table 5 (continued)

Instruction	**What you have coded, advance work done**	**Searching Steps**
Create a cross-tab table. ▪ How many staff at each level did we interview at each site? ▪ Show me the number of focus groups of each type from each site.	▪ Vector nodes set up for site and staff level with mutually exclusive categories. ▪ Both apply to all of the relevant text. ▪ Each document represents an interview.	*Matrix search*: Select the site parent node and the staff level parent node. (The left box will label rows and the right box will label columns.) Choose *Intersect* from the pull down menu in the middle.
Count based on coded themes. ▪ How many subjects said that they are responsible for sanctioning and consider it to be an important part of their jobs? ▪ How many sites had dropped collaborative members? ▪ How many focus groups talked about child care as an important social support?	▪ You have a node that codes these themes (they are responsible for sanctioning, they have dropped collaborative members, etc.). ▪ Each subject is a document or text unit.	*None necessary*, just click on the node and see how many documents are coded to it.
Count based on cases. For example, in how many sites did they implement time limits?	▪ Mutually exclusive vector node set up for site. ▪ Node for “implemented time limits.” (This node has to exclude text about time limits that, for example, documents why they didn’t implement them.)	*Vector search:* Select the time limits node (for the left box) and the site node (for the right box). Choose *Intersect* from the pull down menu in the middle, and click ok. Then export a vector table with “1 for coding, 0 for none” and add up how many boxes had 1’s in them.
Find information about a particular topic that hasn’t been coded yet. We didn’t ask the caseworkers about sanctioning directly, but can you find information on that?	▪ There is no node for “sanctioning.” ▪ There is a node that covers all text from interviews with caseworkers. ▪ You have a good sense, based on reading the data, of the terms teachers use to talk about the school board: “sanction,” “cut off,” etc.	▪ The most reliable thing and the most time consuming would be to read through all of the caseworker interviews, and code the text about the sanctionning. ▪ You can do a text search, and if you have a really good list of words or phrases, it can sometimes be a reasonable approximation. ▪ You would then want to do an intersect with the text search node (sanctioning) and the node for caseworkers.
Find people who said one thing in one part of an interview and another in another part. ▪ Show me the sites that mentioned dissatisfaction with a collaborative member and had never dropped any collaborative members.	▪ Nodes coding the different responses about satisfaction with collaborative members and dropping collaborative members. ▪ Each site has one document that includes information for both questions, though the responses are in different sections of the document.	*Near search:* Select the two relevant nodes for the two boxes at the top of the window. Under “context” choose “document.” Under “retrieve” you can choose first, second, or both. For this immediate purpose, it doesn’t matter, but if you want to see this group’s actual answers to both questions, then choose “Both.” To see how many people are in this group, just click on the resulting search node to see how many documents are coded to it.

    - These aren't necessarily the *only* ways the project could be set up, but they're definitely the most direct ways and in some cases the only way.

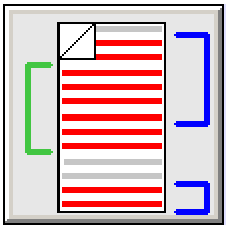
    - For example, N5 can only count documents and text units, so if you want to count caseworkers, and you have more than one document for each caseworker, your counts will be wrong unless you work around this.[5]

    - The key thing is to think about what kinds of questions you're going to want to answer, so that you can set your project up to make it easy to answer that.

  c. For each project set-up in the second column, the third column tells you what kind of a search you would need to do to answer the question in the first column. Some times you wouldn't need to do any search at all.

4. Commonly Used Search Operators: N5 has many more search operators than these, but these are the basic ones that you will use most often. The others, while interesting, are more useful for detailed conversation analysis and other strategies for using qualitative data. For more topic based research, that we do here, these will get you pretty much everything you'll need. For more information on other search operators, see Chapter 8 "About Node Search" in the manual *Using N5 in Qualitative Research*.

  a. Union:

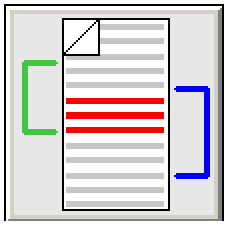
    - 

    - The union operator dumps everything coded to any of a list of nodes into one new node. This is like an "or" command: give me the text if it's in this folder, or in this one, or in this one, or any of these. It also includes text that is filed in more than one of the list of folders (nodes).

[5] If you have documents from multiple waves of interviews, and you have codes for which document refers to which wave, and within each wave each caseworker only has one document, *then* you could set a restriction to only look at one wave of interviews. But then you still have to remember to set the restriction each and every time.

- For example, in Jobs-Plus you have some information on resident participation in the templates and other information in field notes and other materials. Further, in the templates, there are several different questions on resident participation that are all coded to different nodes. So, to bring together all of the information on resident participation from the templates and other materials, you would use a union search on the relevant nodes from the template and the one that contains resident participation information from the other materials.

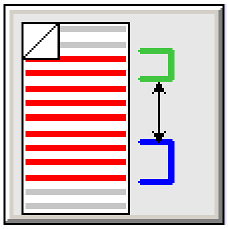
b. Intersection:

- 

- The intersection operator only selects text that is coded to (filed under) *all* of the nodes (folders) in the list. This is like an "and" command.

- So, for example, tell me what the caseworkers in Cleveland said to the question about enforcing participation requirements (sanctioning). You want text about sanctioning, but only the text that is also filed under Cleveland and caseworker.

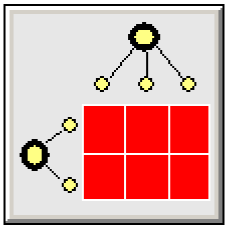
c. Near

- 

- Near is sort of a compromise between intersect and union. Union just throws everything in, and intersect only includes the text units that are literally coded to all of the nodes. Near let's you say, "they don't have to be exactly overlapping, but they have to be close" — within the same document (what I have found most useful), or the same section of a document, or within a certain number of text units. What I have used this for most often is looking for where people said one thing in one part of an interview and another thing in another part of the same interview. Or, if you have a longitudinal study, with several rounds of interviews in each document, and documents that represent people, you could use this to find people who said one thing at the first interview and another thing at the second.

- There are two options:
  - For "Context," choosing "document" will almost always be best. You *might* want to use "Section" or "Paragraph" if you have consistently set up your documents so that a section (the text units between two text units starting with an asterisk) or a paragraph (the text units between two blank text units) make sense for a search like this, but I doubt that that would happen very often.
  - For "Retrieve," you're telling N5 which text you want. The options "first" and "second" allow you to choose just the text from one node — the node in the left space is "first" and the one in the right space is "second." Choosing "both" will get you the text from both nodes where they appear "near" to each other as defined under "Context" above. Note that none of these options really get you all of the text shown in the graphic above.
- You could say something like, "show me the text on sanctions for people who said they have caseloads over 750 people." (Now, I'm assuming that the caseloads category would just be coding the actual text on caseloads, not the whole document, because if it did apply to the whole document, you could just use intersect.) Then you wouldn't want to include the text about how big their caseload was, you would just want to retrieve the text on sanctions.
- With the Jobs-Plus example, "Show me the sites that mentioned dissatisfaction with a collaborative member *and* had never dropped any collaborative members," you might want to get the text coded to *both* nodes to see more details about how dissatisfied they really were, maybe why they hadn't dropped anyone, etc.

d. Matrix

- 
- This search is like a crosstab. For example, from Opening Doors:

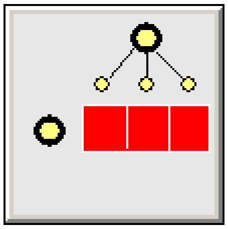
```
Matrix Node: (N 2) //Node Searches/<<Site-by-group>>
Operator: INTERSECT
Description: Search for (MATRIX INTERSECT (1 1) (1 2)). No restriction
Rows: (1 1) /Info-from-Excel/Site
Columns: (1 2) /Info-from-Excel/Focus Group
Data: Number of documents coded

+------------+------------+------------+------------+
| Site       | Current    | Former     | Potential  |
+------------+------------+------------+------------+
| Macomb     |     1      |     1      |     1      |
+------------+------------+------------+------------+
| Valencia   |     1      |     1      |     1      |
+------------+------------+------------+------------+
```

- This search is also like running multiple (in this case 6) searches at once. You could search individually for the intersect of current students and Macomb, and then current students at Valencia, and Former Students at Macomb, and so on and so forth. But this way, you do them all in one fell swoop.

- So, you can actually use any search operator to decide what goes in each cell — union, intersect, near, or a number of others that N5 provides. But for our purposes, it's generally best to use intersect or in some cases near for these.

- Note that in order to use a node for a matrix search, you would choose two parent nodes, with nothing coded to them, and then N5 runs all the searches with the children.

e. Vector

- 

- This search takes a regular node and divides it into categories; it's another way of running multiple searches at once.

- For example, you might want to read all of the sanctioning data from Philadelphia and then read the sanctioning data from Cleveland separately. Doing a regular node report on a vector search node will give you all of the text divided up in that way.

- As with the matrix search, you can use any search operator, but intersect or near are generally best.

5. How to do a node search

    a. On the project pad, click the “nodes” tab at the top, and then the button labeled “Search and Compare Nodes.”

    b. Click on the search operator you want to use. The node search window has buttons that represent what the different operators do.

    c. Different search operators request different information. Fill in the information and click “OK.”

    d. Searches result in a node, which is stored by default under “N — Node searches,” but you can move them to the node tree if you want, by cutting and pasting (“attaching”) the node.

    e. In any case, it’s important to rename the node something meaningful, so that you aren’t stuck with “Node Search 1” and so on later down the road. (Click on the node, then type the new name in the top box, where it says “Search node”, then click on the “Accept Changes” button.)

6. Saving a node search as a command file for later use:

    a. Follow the steps above, but when you have filled in the information for your search, rather than clicking “OK” click “Save.”

    b. You will get a window that says that your search has been saved to the clipboard.

    c. You can either create a new command file (see instructions in Chapter IV) or keep a running log in an existing command file of all of the searches that you’ve done. In any case, open the file, and from the menu choose Edit > Paste, and you will have the command for the search pasted into the command file. It makes sense to put in a few comment lines (any line starting with a semicolon — ;) explaining what that particular search operator produces and the date on which you first ran it, in case you make any major changes to the tree structure later. If you change the tree structure, your old search commands may not work anymore.

7. How to do a text search:

    a. Click on “Search Text of Documents” under the documents tab in the “Project Pad” window.

    b. You can do more complicated searches by clicking the “use special characters” box. For more information on special characters and more complicated searches, see *Using N5 in Qualitative Research*, pages 70-71.

    c. Your search will be recorded as a text search node, and you’ll have to rename it from the node browser window. (Click on the node, then type the new name in the top box, where it says “Text search node”, then click on the “Accept Changes” button.)

    d. Spreading coding: When you do a text search, N5 will automatically code the text units containing the text that you searched for to that text search node. But, especially when a text unit is just one line of text, you might want to see more about the context around that text unit. For more information, see Chapter IV.

8. Saving text searches as command files:

    a. Unfortunately, N5 doesn’t have the same handy write-the-command-for-you capabilities for text searches as for node searches. Here is information on the text search command to use in writing your own command files.

    b. Here is an example of a text search command for any of several phrases.

```
(search-text “[(school board)|politicians|(board of ed)]”
 pattern-search? yes
 node (400 10)
 node-title “Board-of-ed”
 )
```

    c. Node and No-Title indicate where you want the search results to be coded.

    d. You need to say “yes” to pattern search if you are searching for more than one phrase at once. Alternatively, you could do a more simple search that would look like this:

```
(search-text “school board”
 pattern-search? no
 node (400 10)
 node-title “Board-of-ed”
 )
```

e. By default, N5 will code only the text unit with the phrase you're searching for, but in a command file, you can build the "spread coding" into the text search. With "spread-amount you can indicate how much text you want to search to pick up, with the same options as you see when you use the Spread Coding menu option (See Chapter IV for more information). Just add one of these clauses to your command.

   - spread-amount to-document

   - spread-amount to-section

   - spread-amount to-paragraph

   - spread-amount 5 (if you just give a number, N5 will interpret that as a number of text units you would like to retrieve on either side of the phrase)

f. If you want to only search interviews with teachers, and those interviews are all coded to node (1 1), then you could use the set restriction command to avoid searching other documents:

```
(set-restriction only document (1 1))
(search-text "[the board|school board|those politicians]"
 pattern-search? yes
 node (400 10)
 node-title "Board-of-ed"
 )
(end-restriction)
```

E. Reports on Matrix and Vector Nodes

1. Unfortunately, you can't browse a matrix or vector node. But, you can create a regular node report, using the same steps as above, and N5 will separate out the text by grouping within the matrix or vector. So, if you have a vector search that is the node for sanctioning information intersected with the nodes for site, and you produce a regular node report, you will get the sanctioning information listed by site.

2. Exporting a table: You can also use matrix and vector searches to produce tables with numeric output.

   a. Make sure that your numbers will mean something. Refer back to Chapter III and defining text units and documents.

   b. See above and Table 4 about conducting matrix and vector searches.

c. From the menus, select “Project,” “Export Table,” “Matrix/Vector…”

d. In the window that comes up, select the matrix/vector search node.

e. Choose an option for the cell display

- 1 for coding 0 for none: Pretty self-explanatory, it’s a yes or no indicator. In Opening Doors, did all types of focus groups talk about financial aid, or only the current and former students?

- Number of text units coded: If your text units are meaningful, (as in Opening Doors, where a text unit is a rough approximation of a mention in the focus group) you might want to count them.

- Number of documents coded: Same for documents — in Opening Doors, you might want to know how many focus groups included discussion of a particular topic.

- Number of text units per document: This report is not so helpful, for the kind of work that we do, because you can only get it for one document at a time.

f. Choose whether you want to see the report on your screen, or save it directly to a text file. (If you look at it on screen first, you can still save it as a text file later. N5 will prompt you to save the report when you go to close the window.)

F. Fact-Checking

1. Note that the big caveat applies here as well. The simple fact of using N5 does not absolve authors of responsibility for documenting their analysis. It can simply facilitate documentation if the project is set up appropriately — so that the items you count in your writing are represented by documents and/or text units, and the categories you count and/or refer to in your writing are accurately, consistently, and clearly represented by nodes.

2. See the qualitative fact-checking guidelines (Appendix B) for more information on guidelines for documenting and checking findings drawn from qualitative data. Many kinds of facts drawn from qualitative data (e.g. program descriptions) will still need to be checked and verified by the authors. However, some kinds of facts — specific counts and quotations — can be more easily documented and checked in N5.

3. For example, we might want to verify the following text in a report about the Urban Change Implementation data:[6]

> Administrators generally supported time limits. For example, one from Cleveland said, "I personally like the time limits. It has been a mindset change for our workers at large. It has been a continuous education process for our clients; they have to really think about how to support their families. I think by putting time limits in place, it forces us to deal with self-sufficiency much sooner, whereas we did not always in the past. Clients really have to focus on what type of provider they want to be for their families."

   a. In checking the first sentence, we would see that there is no node for "support time limits," so we would check for other documentation (e.g. a list of the document names of interviews with administrators who supported time limits) and if we couldn't find that within two minutes, we would just mark off the sentence and show it to the author for further documentation and/or sign-off.

   b. The next sentence is a quote, so we would find it using a text search, verify each word in the quote, make certain that the quote wasn't taken out of context, and check that the quote is also coded to "Cleveland" and "Administrator."

4. Verifying quotes:

   a. Make sure that the base data referred to in the text you're checking (e.g. site, staff level, etc.) is coded and applied to all of the relevant text, in this case to full documents.

   b. Using the Project Pad, choose "Search text of documents" under the Documents tab. Or, from the menu, choose "Documents," "Text search," "All documents."

   c. Then type in some words from the beginning of the quote you're looking for.

      - N5 will look for a string in one text unit, so make sure that the text you type in is not likely to be broken across text units. There is less risk here if you have longer text units, I usually start with searching for 3-4 words maximum to be safe, if you find a ton of text units with those three words you can add on another to narrow the search. (See Chapter III for more information on defining text units.)

      - With the quote above, I might start with "personally like the time limits". (Don't type the quotes though, unless you want N5 to search for quotes.)

[6] This text does not actually appear in any Urban Change report.

d. A report window will pop-up, and you can just close it.

e. Click on the resulting text-search node in the node explorer, and rename the node to indicate what you were looking for. Click on “Browse”, right click on the quote, select “jump to document” (to make sure that it is really the full quote you’re looking for, and that the context in the document is what the statement describes), and then right click on the quote and choose “Examine Coding” to make sure that the text is also coded to “Cleveland” and “Administrator”.

5. In addition to the searches and reports mentioned above, you can just click on a node in the node explorer to see the number of documents and text units coded to it, which may help with checking some quantitative statements.

VI. Sharing Data on a Team

A. What Can We Learn from Quantitative Data Managers?

1. Follow naming standards. Each team can agree on these, and they will vary from team to team, but it helps immensely to facilitate communication if everyone follows standard practices in naming documents, nodes, projects, and reports. The one thing to keep in mind in developing standards is that simpler standards are much easier to follow.

2. Avoid file proliferation. It's best to keep things simple — that is, one version of the data set is much easier to keep track of than five. That may not be practically possible, but still five is easier than ten.

3. Back up data: Data stored on the network is automatically backed up by the help desk, so it makes sense to have a central data place on the network where raw files (word documents) and N5 projects will be stored.

B. Ideal Situation

1. One person, the data manager, has the N5 project. (The data manager is a research assistant or analyst working half-time on managing qualitative data.) S/he is in charge of keeping track of the documents, importing them into N5, entering coding (doing minimal interpretive work), and producing reports requested by the authors. S/he stores the data on the network, which help desk staff back-up at least once a week.

2. Anyone who collects data for the project sends final versions of written materials to the data manager. And because people follow naming conventions and keyword coding, and because the data are structured, the data manager does basic information and topic coding consistently and accurately.

3. The data manager provides any information on the topics requested by the authors, and then authors do further analysis on paper, and the data manager enters additional coding into N5 as needed for fact-checking documentation.

4. This is ideal because: everyone has easy access to the data they need; there is only one version of the project (not multiple versions floating around); the coding process is easy and

painless, because of consistent and high quality keyword coding/structured documents; all documents are accounted for; and the dataset is backed up regularly.

C. In Reality…

1. Although this might be ideal, things don't always work quite this way. Below are some alternatives to this situation and suggestions for working with each of them.

2. How many of the authors want to use N5, meaning it's relatively easy for them, and they think it will be efficient?

   a. If none, best to code on paper and have it entered in N5 and then get reports. Just have data manager do basic coding and reports. This way, there is only one copy of the project, which means no file proliferation.

   b. If some or all, then they might save time coding in N5 as they read and analyze the text. Alternatively if there is too much coding happening for one data manager to enter it all in a timely fashion, it might make sense for authors to enter it themselves. If so, the data manager should give each author their own node to work off of, and periodically (every few weeks or whenever a lot of new data come in) merge projects. Alternatively, authors may not want to do coding on screen, but might want to get reports more quickly by creating them themselves; this might be best done by accessing the current copy of the project on the network (and consistently naming any searches that they want to keep) or periodically sending out updated versions of the project to team members to use on their own computers, with the understanding that they will need to keep track of document names and text units for fact checking, because searches will be lost in new versions of the project.

3. How structured are the data?

   a. The more structured the data are (e.g. write-ups follow an interview guide), the easier it is to have an analyst or research assistant do coding. Even if it's not keyword coded in a way that N5 could automatically file, generally consistent subheadings can be like having a built in guide.

   b. The less structured the data, the more support a junior staff person would need in order to do interpretive work. Authors will need to balance the time needed to support junior staff to interpret things the way you would, with the time it would take you to do the coding yourself. In general, it's only a good idea to ask a data manager to do topic coding if the people writing up the notes used some sort of standard categories or headings or separately addressed a standard set of themes.

   c. Note: Data are almost always structured by site and date and type of document (observation/interview, maybe round of visit, etc), and it is generally helpful for a data manager to code these types of basic characteristics of each document. At the very least, in the analysis phase rather than having to search around for documents, authors will be able to quickly get the documents that they need from a particular site over a particular period of time.

4. What role will consultants play?

   a. If there are no consultants on the project, then don't worry about this.

   b. If consultants are data collectors only, you only need to make sure that incoming documents are formatted and structured as planned, and that they go to the right people (the data manager and a senior project person as back up, particularly when there is risk of turnover in the data manger position). See below on quality control.

   c. If consultants are central partners in analysis and writing, you will need a less centralized approach. In these cases, either e-mail coding templates to the consultants to use when coding hard copies of the documents or give each consultant/author a node to work from, and then periodically have them send their project to the data manager, to perform merges with the MDRC project(s). Do ask the consultants to send new data files to the data manager rather than importing them themselves, so that naming and logging conventions can be followed in house.

D. Management Models

1. Projects use a range of models on centralized versus decentralized control of the data.

   a. Model 1: Very centralized, described above.

   b. Model 2: Basically the same as model 1, but the data manager periodically distributes the project to other team members so that they can create reports for themselves. Some version control is necessary, but you don't have to worry about merging and there is less of a bottleneck problem.

   c. Model 3: All (or more than one) team members code in and produce reports from N5, and the N5 project is periodically merged and redistributed to the team. There is no bottleneck problem, but you have to pay *very* careful attention to merging and version control. There are also many more versions to keep track of (file proliferation) in this model.

E. Some General Guidelines Across Models

1. Always try to keep things simple. Murphy's law applies: if something can go wrong, it will.

2. Always have one data manager (research assistant or analyst) who knows where the data are, brings documents into N5, logs basic information about the documents received, and keeps track of the versions of the N5 project whether there is just one or many. Note that this is probably at least a half-time job.

   a. Create a project network directory that includes:[7]

      - a record of incoming materials (a list of all of the documents received, some basic information about each document such as date, interviewee, and/or site, where they are stored on the network, and their document name in N5 once they have been imported into N5; see Table 6, for example)

Methodological Innovations Initiative

Table 6

Urban Change Institutional Study — Excerpt of the Record of Interviews Received (Cuyahoga County, Glenville neighborhood)

New Filename	Name of Institution	Date of Interview	Site Assigned #	Original File Name	Add'l Material
cgle_allmyheart	All My Heart Education Center	7/10/1998	IN1164112	IN1164112	NO
cgle_ghealth	Glenville Health Center	7/30/1998	IN1165119	IN1165119	NO
cgle_oppcenter	Glenville Neighborh'd Opportunity Ctr	11/3/1998	IN1187130	IN1187130	YES
cgle_plaza	Glenville Plaza Childcare	7/21/1998	IN1164115	IN1164115	YES
cgle_hunger	Hunger Center at Cory Methodist	6/25/1998	IN1182103	IN1182103	NO
cgle_jfselem	Joseph F. Landis Elementary School	1/21/1999	IN1181135	IN1181135	YES
cgle_mariah	Mariah Hope Program	7/10/1998	IN1161113	IN1161113	NO
cgle_north	Northeast Comm. Mental Health Ctr.	7/9/1998	IN1164111	IN1164111	YES
cgle_safeh_M	Project Safe Harbor	7/17/1998	IN1166118	IN1166118	YES
Appended	Project Safe Harbor	7/1/1998	IN1169105	IN1169105	YES
ap_cgle_stmartin	St. Martin de Porres Family Center	11/20/1998	IN1184137	IN1184137	NO

      - a running memo containing decisions about naming conventions and basic coding;

      - the current version of the project; and

[7] Note that you will need to ask the help desk to create the directory for your project.

- a subdirectory containing old versions of the project (if applicable).

b. Do some quality control to make sure that minimal standards agreed on by the team are followed in the data submitted

- This is not about the overall quality of the notes or transcripts, although teams should address those issues as much as possible.
- To the extent that you use keyword coding in writing up the materials, you want to be sure that everyone who will be writing up notes and/or inserting keywords knows the technical details of how to do it, what the codes mean, and when to apply them. The data manager can help by creating notes templates. Authors can help by keeping the keyword coding very simple, and providing training on the meaning of any topic or theme codes.
- Once data comes in, the data manager will need to check that data collectors have used the templates appropriately.
- The data manager will also need to check for formatting issues (bolding etc.) discussed in Chapter III.

3. Good documentation and project directories on the network are the best way to deal with staff turnover.

   a. Having one data manager in charge of the version control and keeping data in one place can be really helpful, especially in terms of keeping track of data collected by staff who may leave over the course of the project.

   b. The only cautionary note is for teams to build in some mechanisms for data managers to transfer their knowledge to someone else, should they leave. Ideally there would be some overlap time between the two people doing this job, but if not, good documentation (descriptions of files and process) will help the new person pick up where the other left off.

4. Always keep (at least a copy of) the N5 project on the network

   a. Files on the network are backed up automatically by help desk staff.

   b. Multiple MDRC staff, though only one at a time and not across NY and CA offices, can access the current project if it is on the network.

F. Using WinZip to E-Mail N5 projects

1. Each N5 project has a parent directory, a file in that directory called project.stp (by default), and then several subdirectories: Commands, Database, Rawfiles, Reports. This means that you can’t simply attach it to an e-mail the way that you would a word file. In order to send a project over e-mail, you need to send all of the files, with the directory structure, so we use WinZip to do that.

2. To send the project

   a. Start WinZip:

      - You may need to click “I agree” to the licensing, in order to get the program started.

      - A window should open titled “WinZip Wizard” if it doesn’t, then click the button titled “Wizard.”

   b. Click the “Next” button to start.

   c. Choose “Create a new zip file” and click next.

   d. WinZip will ask where to save the file, and this will be the file that you e-mail, so tell WinZip to save it somewhere that you will e-mail from. (You may need to use the browse button to select a directory.) Name the file the same as the N5 project name.

   e. Click the “Add folders” button and select the N5 project parent directory. (The folder that contains the project.stp file and all of the N5 subdirectories, but not the files themselves.)

   f. Click “Zip Now” and exit WinZip.

   g. In GroupWise, you can attach the zip file you created above in the same way that you would attach a word file or anything else to an e-mail.

3. To receive/unzip the project

   a. From Groupwise, you can double click on the zip-file icon in your e-mail or save it on your hard drive and find it from the “My computer” icon on your desktop and double click on it there. (Note: You may get a window saying that WinZip is not registered on your computer. Just click “I agree” and it will be fine.)

b. If a window opens entitled “WinZip Wizard” then you are all set. If not, click the button entitled “Wizard” on the tool bar.

c. Click on the “Next” button, choose “Unzip or install from [name of the .zip file in the e-mail],” and click “Next” again.

d. Click on the “Select different folder...” button and choose where you want the N5 project to be saved.

   - Note that N5 is set up to automatically look for projects in the directory below.

     ```
     C:\QSR Projects\N5\Projects\
     ```

   - You may want to put the project there, to make it easier when you start N5. (You won’t have to use the “Browse” button.) *But, if you do this, don’t forget that you are storing data on your ‘C’ drive, make sure that there is a copy on the network for back-up purposes.*

e. The option “Display file icons after unzipping” should be checked. (Select it if it's not.)

f. At the bottom of the window, click the button labeled “Unzip Now.”

g. Once it is finished you will get a message which says, “The unzip operation is complete.” The Wizard will open an Explorer Window showing you where the files were unzipped to on your computer.

h. Click the “Close” button to exit WinZip.

G. Merging N5 Projects

1. The instructions below apply to merging two projects, but you can just keep merging additional projects (run through the process as many times as necessary) to merge more than two projects into one.

2. Checklist for preparing projects to be merged:

   a. For more information on the logic that N5 uses in merging projects, see the section below. These guidelines are based on that logic.

   b. Make sure that no changes have been made to any documents within either project: meaning, nothing new appended and no editing of the text of the documents or the document descriptions themselves. If something has been appended in one project, then append it in the other as well, so that the document will still be read as the same.

   c. It's fine if completely new documents have been added to one of the projects. For version control purposes, you may want to alert team members to this change though — "Hey, don't forget to look at documents A, B, and C, appearing for the first time in this version of the project!"

   d. Make sure that the nodes in each project mean the same thing: meaning, if anyone has added a new node or nodes to their project they added them at addresses that are not in use in the other project. If each person is working from their own base node, and has not deleted any nodes that were in the prior version, then this should not be a problem.

3. Once you have addressed the issues above, merging N5 projects is incredibly simple.

   a. N5 comes with a separate program called QSR Merge that merges N5 projects.

   b. QSR Merge will run fastest if you have copies of the two projects on your actual computer, and it's a good idea to make extra copies of the input projects before your merge them anyway, but you can merge projects located on the network if necessary.

   c. QSR Merge is actually a separate program from N5, so exit N5 and open QSR Merge.

   d. Choose "Merge Project" in the window that pops up.

   e. Click on "Input Project 1" and select one of the projects you intend to merge; click on "Input Project 2" to select the other. (Assuming that you followed the guidelines above, it won't matter which project is #1 and which is #2.)

   f. Click on "Output project" and choose the folder where the output project should be located. The name you type in to the "filename" box will be the name of the parent directory of the project, so don't create a new directory for the project beforehand. Follow naming conventions for naming this folder, so that you can distinguish it from the other projects. Also, do not make this a subdirectory of any other project.

   g. Click on "Merge," and that's it.

4. Post-merge clean-up

   a. If you have been using memos, then go through all of the memos and delete the text that is in there twice.

b. Copy files from the “command files” and “raw files” directories of the old projects into the new project. (You may also want to copy files from the “reports” subdirectory, but there will probably be many of them, and they are much more easily re-created.)

c. Delete any duplicate search nodes.

H. The Logic of Merging N5 Projects

1. Merging data documents: N5 defines documents as the same if they have the same name, they have the same number of text units, and the section dividers show up in the same text units. This is important because we want coding that one person does on a particular document to be applied with the coding that another person has done on the same document; we also want the coding that I did on my version of a document to apply to the actual text that I coded, and not to some other text. This will all be fine if you do not edit or rename any documents once they have been imported into N5. However, you might want to append information to documents (e.g. either additional interviews with the same staff person). That’s fine, just be sure to merge all of the relevant versions, and then do the appending and redistribute the merged project.

2. Merging memos: N5 appends everything in the first project’s memo (for a particular document or node or whatever) to the second project’s memo (for that same document or node or whatever). This means that if you do multiple merges, the memos will grow exponentially in length, without adding much, if anything, in content. This is one reason why I prefer not to use memos. But to rectify it, you can go through and delete extra text from memos in a new project after you do the merge.

3. Merging codes (index systems): If a document occurs in both projects that are being merged, the actual document itself will be copied from one of the projects. However, all of the coding for that document — from both projects — will be merged. So, say, for example, that Node (3 4 5) occurs in both projects, and the identical document “Mary” does too. Suppose the indexing for “Mary” at (3 4 5) is: Project 1: [lines] 4-9, 22-25, 36; Project 2: [lines] 1-3, 11-16, 24-28, 36-38. Then the merged coding is just what you would get from the Union index system search operation: Merged project [lines]: 1-9, 11-16, 22-28, 36-38. If there is a node address that is only used in one of the projects to be merged, then it and its indexing just get copied as is to the output project. But beware, because if node (3 4 5) is “Cleveland” in Project 1 and “Child care policy” in Project 2, the nodes will still be merged in this way if they share an address, and in the new project, the node will just be renamed “Cleveland”.

4. Important pitfalls in merging N5 projects:

   a. The example above assumes that node (3 4 5) has the same meaning in both projects, but if they don't actually have the same meaning, then there will be two separate codes treated as the same and that will make node (3 4 5) in the merged project meaningless. In order to avoid this pitfall, either allow only the data manager to create or delete nodes, or give other team members their own node off of which they can create new nodes, but not delete any.

   b. QSR merge does not include any command files or raw files (such as original documents or spreadsheets) from the old projects in the new merged project. However, you can manually copy them from the old projects into the new one once you have run the merge program and/or keep copies of older projects and refer back to them there.

   c. Search nodes are not merged, they are copied separately in the new project, and if you merge multiple times, this can result in the same problem as the one with memos. In order to avoid this, do searching in command files (use the search-nodes and/or search-text command discussed in Chapter V), and then you can delete the search nodes before merging and just run the command files on the merged project.

Appendix A

Spelling Checker

(Anonymous)

I have a spelling checker
I disk covered four my PC.
It plane lee marks four my revue
Miss steaks aye can knot see.

Eye ran this poem threw it.
Your sure real glad two no.
Its very polished in its weigh,
My checker tolled me sew.

A checker is a blessing.
It freeze yew lodes of thyme.
It helps me right awl stiles two reed,
And aides me when aye rime.

Each frays comes posed up on my screen
Eye trussed too bee a joule.
The checker pours o'er every word
To check sum spelling rule.

Bee fore wee rote with checkers
Hour spelling was inn deck line,
Butt now when wee dew have a laps,
Wee are not maid too wine.

And now bee cause my spelling
Is checked with such grate flare,
There are know faults in awl this peace,
Of nun eye am a wear.

To rite with care is quite a feet
Of witch won should be proud,
And wee mussed dew the best wee can,
Sew flaws are knot aloud.

That's why eye brake in two averse
Because Eye dew want too please.
Sow glad eye yam that aye did bye
This soft wear four pea seas.

Appendix B

Qualitative Fact Checking Guidelines

MDRC

16 East 34th Street, New York, NY 10016-4326
(212) 532-3200 Fax: (212) 684-0832 www.mdrc.org
Regional Office:
88 Kearny Street, Suite 1800, San Francisco, CA 94108-5545
(415) 781-3800 Fax (415) 781-3820

memo

To: Barbara Goldman
From: Tom Brock, Vanessa Hosein, Sue Scrivener, and Anne Sweeney
Date: July 11, 2000
Re: Recommendations for Fact-Checking Qualitative Data

During the past six months, we have been working to develop fact-checking procedures for qualitative data in MDRC reports.[1] Although most MDRC publications contain data gathered through qualitative methods — including interviews with staff and clients, observations of programs, review of procedures manuals and other program documents, etc. — to date there are no clear guidelines for making sure qualitative data are reported accurately. The procedures used to verify quantitative data, such as reviewing SAS printouts and cross-checking numbers on tables to those in the text, do not always translate to qualitative text. Some research teams have tried to compensate by asking research assistants to sift through field notes and other original source documents to confirm what authors write, or having report authors "sign off" on blocks of text. Nonetheless, the lack of formalized procedures has left many authors and fact-checkers feeling frustrated and uncertain about whether they are going too far or not far enough to verify information.

Our working group was formed to develop a process for fact-checking qualitative data. Specifically, we have been looking to develop a set of written procedures that will uphold MDRC's commitment to accurate reporting of data while also clarifying — and, we hope, simplifying — the work of report authors and fact-checkers. After much thinking and discussion, we reached agreement on four core principles:

1. *Not all qualitative "facts" are created equal.* Some of the data we gather through our qualitative research (for example, a description of a policy that we are studying) are more central to our analysis and conclusions than others (for instance, an illustrative case study of one person's experience in a program). We suggest that the

[1] We want to acknowledge helpful input from Barbara Goldman, Dan Bloom, Barbara Fink, Janet Quint, and Ana Ventura.

level of effort that goes into fact-checking qualitative data be adjusted accordingly.

2. *Qualitative data sources should be documented and available for review.* To the extent possible, authors should keep track of where their information comes from, whether it be published material, field notes, or other sources. Fact-checkers and other reviewers should be able to examine the sources to understand how the author reached his or her conclusions or determined what to include in the text.

3. *Everyone on a report team has a role in the fact-checking process.* Although one or more research assistants are typically designated as fact-checkers, we believe that authors, consultants, report coordinators and reviewers contribute to the fact-checking process. All of these people should have a clear understanding of the work that that they and others are contributing to avoid duplication of effort and increase confidence in our reporting.

4. T*he buck stops with authors.* While fact-checking provides an important measure of protection against error, authors are ultimately accountable for the accuracy of all material contained in their papers or report chapters.

Inevitably, our discussion about fact-checking procedures touched upon other issues related to qualitative research. For instance, how do we know whether we can trust our field research? What are our expectations for writing up field notes? We also got into issues about what we should expect of consultants who collect qualitative data for our studies. While these issues fall beyond the scope of this memo, we hope that they will be addressed by future working groups.

The rest of this memo is organized into three sections. Section I defines seven different kinds of qualitative facts and provides examples of each. Section II presents our views on the fact-checking roles and responsibilities of the various people involved in the production of reports and other documents, and Section III briefly describes the general procedures for fact-checking and presents guidelines for checking each kind of qualitative fact.

Section I: Kinds of Qualitative "Facts"

One of the challenges in developing fact-checking procedures for qualitative data is that there are many different kinds of qualitative facts — and what seems appropriate for one type of fact may not be appropriate for another. We identified seven types of qualitative facts that commonly appear in MDRC reports: statements (typically derived from field interviews or observations) that use quantitative terms or modifiers; quotations from published sources or interviews; descriptions of laws or policies; descriptions of program operating procedures; vignettes or profiles of individuals or families; overall summaries or concluding statements; and statements of general knowledge. Below is a brief explanation of each, followed by examples from MDRC publications.

Statements that use quantitative terms or modifiers

⇒ Any statement that indicates or implies numbers, proportions, ratios, frequency, etc. (See Attachment A for a list of commonly used quantitative terms or modifiers that should be fact-checked. Precise numbers should be fact-checked according to the quantitative fact-checking standards.)

Examples:

> "*By and large*, women felt that welfare reform was positive. They accepted the mainstream public's view that *many* welfare recipients were wastrels and frauds…." (*Big Cities and Welfare Reform*, p. 167.)
>
> "As these descriptions suggest, *all* of the states allow only a *few* up-front exemptions from the time limit." (*Implementing Time-Limited Welfare*, p. 38.)

Quotations

⇒ Any information taken verbatim from a published source, an interview, or another source.

Examples:

> "William Julius Wilson, in his widely cited book *When Work Disappears*, writes that 'for the first time in the twentieth century most adults in many inner-city ghetto neighborhoods are not working in a typical week,' adding that 'the current levels of joblessness in some neighborhoods are unprecedented.'" (*Mobilizing Public Housing Communities for Work*, p. 2.)
>
> "A central office staff member acknowledged that workload issues affect staff views: 'Staff are not philosophically opposed to the premise of welfare reform. But it is an overwhelmingly heavy burden.'" (*Big Cities and Welfare Reform*, p. 167.)

Legal or policy descriptions

⇒ Descriptions of formal policies and procedures based on legislation, policy manuals, or field research.

Examples:

"Additionally, in keeping with its commitment to providing services to victims of domestic violence, CalWORKs gives benefits for up to six months to immigrants who are victims of domestic violence and who are eligible for special residency under the federal Violence Against Women Act (VAWA). This provision targets women who have initiated the process of gaining citizenship but are prevented from completing it because they have ended their relationship with their legal sponsor when that person is also the perpetrator of abuse." (*Big Cities and Welfare Reform*, p. 87.)

"To encourage and reward work, all earned income is disregarded (that is, not counted) in calculating recipients' cash grants (and Food Stamp benefits) as long as their earned income is below the federal poverty level. Recipients become ineligible for cash assistance if their earnings are at or above the poverty level." (*Jobs First: Implementation and Early Impacts of Connecticut's Welfare Reform Initiative*, p. ES-6.)

Program operating descriptions

⇒ Descriptions of organizations, operations, staff-client interactions, etc., usually derived from field research. Program operating descriptions differ from policy descriptions in that typically they require the author to translate what researchers have heard and observed into a written explanation of "how things work." Stated another way, policy descriptions describe policy makers' intentions; program descriptions describe what is actually taking place.

Examples:

"Across the demonstration cities, welfare departments and other agencies have begun to outstation case managers at the Jobs-Plus developments and, more generally, to work out procedures for referring residents to the agencies providing employment-related services. This has required that the collaboratives begin sorting out how to coordinate their efforts in order to avoid duplicating functions…. In another form of service integration, several sites are developing a common intake form that a variety of agencies will use, as a way to reduce some of the burden on residents involved with multiple agencies." (*Mobilizing Public Housing Communities for Work*, p. 31.)

"At another site, MDRC staff observed an intake interview. It was polite but relatively brief and centered on getting the information needed to complete the enrollment form." (*New Chance: Implementing a Comprehensive Program for Disadvantaged Mothers and Their Children*, p. 72.)

Vignettes and profiles

⇒ Brief summaries of specific cases or situations as illustrative examples.

Example:

> "Jackie, a 35-year-old African-American woman, talked about how her four children are doing in school, and her concerns and worries about her sons compared with her daughters. Like the great majority of parents in the study, she is concerned about her children's schooling, safety, and future lives." (*New Hope for People with Low Incomes*, p. 194.)

Summary and concluding statements

⇒ Statements that depict overall impressions or draw conclusions from a wide body of evidence.

Examples:

> "Overall, then, implementation of the structural elements of the program model was less robust than it might have been. And while some Project Transition teachers altered and expanded their instructional practices, especially in response to the introduction of more integrated curriculum units, the project does not appear to have been as much of a catalyst for instructional change as some of the parties to the demonstration had hoped." (*Project Transition*, p. 15.)

> "There is one overarching lesson from this early period of New Chance operations: It has proved possible to implement the first phase of the model in diverse settings, to recruit a highly disadvantaged group of young mothers, and to create a program that yields satisfactions for participants and staff alike." (*New Chance: Implementing a Comprehensive Program for Disadvantaged Mothers and their Children*, p. 199.)

> "The 'core' benefits and services that New Hope offered — the community service jobs, earnings supplements, health insurance, and child care assistance — were delivered by New Hope staff to qualified participants as the program designers intended…. If there was an overarching problem with the delivery of benefits and services, it was that many participants seemed to have difficulty understanding fully how the process worked." (*Creating New Hope*, pp. 161-5.)

Statements of general knowledge

⇒ Statements that contain general information that was not taken from a specific source or sources; sometimes "common knowledge" within a field.

Examples:

> "Welfare reform has been near the top of the American political agenda for almost a decade, a reflection of persistent dissatisfaction with the Aid to Families with Dependent Children (AFDC) program." (*Evaluating Two Welfare-to-Work Program Approaches: Two-Year Findings on the Labor Force Attachment and Human Capital Development Programs in Three Sites*, p. ES-1.)
>
> "The early 1990s began as a flashpoint for welfare policy but soon became a watershed." (*Encouraging Work, Reducing Poverty: The Impact of Work Incentive Programs*, p. v.)

Section II: Fact-Checking Roles and Responsibilities

Every MDRC report represents a team effort, involving a large group of authors, research assistants, reviewers, and sometimes consultants. Although one or more research assistants may be officially designated as the "fact-checker(s)," one of our premises is that *everyone* on the team has a role in verifying that the information contained in a report is accurate and supported by evidence.

Report authors are ultimately responsible for the information contained in a report. Just as staff who write about impacts or costs must be able to show how they derived their figures, staff who write about qualitative data must be able to produce the field notes or other materials they used to describe a program, characterize people's experiences, etc. Therefore, we recommend that authors keep careful records of the data sources they use, and, at the beginning of the fact-checking process, provide fact-checkers with a packet containing these sources. This is not as arduous or time-consuming a task as it sounds (Attachment B provides some tips on how to do this). We also propose, in the following section, that in some instances, it may be more efficient for authors to fact-check some information themselves, rather than to provide original source material to a "fact-checker" and ask her/him to retrace the author's steps.

Consultants who contribute qualitative data to our reports, such as site-based field researchers, must be held accountable for the information they submit to us. We do not think that they necessarily should be required to give us all of their field notes or other background documents (although we may push for this in the future), but they should document their sources with the same degree of care we expect of authors. Consultants should also agree to provide copies of relevant source material on request. We suggest that in the future, consultant contracts spell out these obligations clearly, and that project directors or lead authors review the need for proper documentation with consultants at the beginning of a project and again before any deliverables are submitted. Consultants who are authors of reports should be held to the same standards as MDRC staff.

For each MDRC report, at least one research assistant is designated to be a **fact-checker**. Fact-checkers are expected to read through all text and tables carefully, bracketing information that needs to be verified, checking this information against source material or following other agreed-upon procedures, and marking off each fact that has been checked. When the fact-checking process reveals errors or inconsistencies, the fact-checker should consult with the author(s) to resolve them. Finally, the fact-checkers should make sure that report standards are followed and that there is internal consistency within the chapter(s) they are reviewing.

Report coordinators oversee the fact-checking process, making sure that all chapters are reviewed according to procedures and resolving any questions that arise about which items need to be fact-checked, consulting with authors as necessary. (Sometimes report coordinators fact-check some material themselves.) Report coordinators are also expected to generate the report standards, including a list of key words and phrases that should be used consistently throughout a document. After the fact-checking process is completed, report coordinators should maintain at least one copy of the fact-checked manuscript within MDRC, either with a project team member or in central files, and send another copy to be archived off-site.

Report reviewers already play an important role in checking both the content and tone of reports issued by MDRC. We are not advocating any change in the role that reviewers play, but rather want to recognize the contribution they already make to the fact-checking process. In particular, reviewers help determine whether the conclusions drawn from qualitative data — for example, that a program received a fair test (or did not), that staff and participants understood program rules and benefits (or did not), etc. — are justified based on the data presented. An important part of the reviewers' function is to ask report authors to review their source material, provide more evidence, and revise their chapters when the information presented seems unclear, unconvincing, or inaccurate. At least one reviewer should be designated to provide a critical read of the *whole* report — including qualitative *and* quantitative chapters — to make sure that the overall story makes sense and that important themes and terms are consistent throughout the document.

Section III: Qualitative Fact-Checking Procedures

This section briefly describes the general procedures that we recommend to fact-check qualitative information in a report, and then presents recommended guidelines for checking the different types of qualitative facts described above. In addition, the end of the section discusses the process of reading chapters and reports for internal consistency.

General Procedures

As recommended in the revised report coordination materials, authors (and consultants) should indicate their sources in the chapters they write. As Attachment B describes in more detail, sources can be noted in endnotes, annotations, or bracketed in the text. Authors should also provide the relevant source material to the fact-checker in a fact-checking packet. Examples of source material include relevant pages from field notes, tick sheets, NUD•IST output, and photocopies of published documents. (See Attachment B for hints and tips for authors and consultants.)

Ideally, information should only be fact-checked once; for text, this works best after the document has been reviewed and edited. In the past, fact-checkers have found it works well to go through the material four times. The first time is just a quick perusal to get a general idea of the topic and the report or chapter organization. In the second read, the different types of facts can be identified and bracketed so that the fact-checker can easily distinguish what s/he is responsible for checking and what s/he will not be checking. During the third read, the fact-checker can verify each fact using the source material provided in the fact-checking packet. While doing this, the fact-checker should follow the "two-minute rule" presented in MDRC's general fact-checking materials: If a fact-checker is faced with a fact that will require more than two minutes to verify, s/he should circle it for the author either to sign-off on or provide materials for the fact-checker to verify the fact. In the fourth read, the fact-checker can check internal consistency within the chapter and verify cross-chapter references ("As noted in Chapter 3,…"). At this stage, the fact-checker can also review the entire chapter to confirm that all identified facts in the chapter have been accounted for, and that all fact-checking issues are resolved.

After the material has been verified, the fact-checker should indicate this (with check marks) on the fact-checking copy of the document. The source used to check the material should be noted in the margin. For example, when checking a program description based on field notes, the fact-checker should note the page number(s) or the line numbers from which the material was taken. When a fact-checker finds a discrepancy between the material being checked and the source, the questionable material should be underlined and brought to the author's attention. If a correction is necessary, once the change is made the new material must be checked and attached to the original fact-checking draft.

Checking Different Kinds of Qualitative Facts

In developing processes to check different kinds of qualitative facts, we tried to strike a balance between reasonableness and rigor. In contrast to common past practice, we do not think that every statement should be verified by the fact-checker, but rather that responsibility be shared between the fact-checker and other team members. We divided verification of facts into three categories: verification involving fact-checkers and authors, verification involving reviewers and authors, and verification involving only authors. We present our recommendations below and summarize them in a table (see Attachment C).

1. Verification Involving Fact-Checkers and Authors

- **Statements that use quantitative terms or modifiers**

Quantitative terms (e.g., "all," "most," "few") should be accounted for by the author and verified by the fact-checker. We propose that whenever such terms are used, authors should provide a "summary" of the evidence, rather than the original data source; fact-checkers should *not* have to pour through pages and pages of field notes. Authors should provide documentation, such as tick sheets or annotations, that justifies the word or phrase in question (e.g., "most" = 7 out of 10 staff interviewed). Authors should use their judgment in documenting the quantitative terms they use. For example, if a sentence says, "The majority of staff expressed anxiety about the upcoming program changes," it is only necessary to document that more than half expressed this emotion, not to account for every staff member's feelings.

Using the evidence provided by the author, the fact-checker should confirm that the actual numbers correspond to the stated or implied quantitative description. (Teams may set standards for what qualifies as "most," "few," etc., and report coordinators should provide these standards to fact-checkers.) Above each quantitative term or modifier, fact-checkers should write the actual numbers on which the word is based.

- **Quotations**

MDRC would fast lose credibility if we misquoted our sources. For all quotations of published material (i.e., journal articles, books, etc.), authors should provide references with page numbers or photocopies of published materials from which quotes are drawn. The fact-checker should compare each word in the quotation to the source, place a check mark above each verified word, and note the source in the margin. The fact-checker should also check to see that the quote is not taken out of context.

Report teams should decide whether or not a fact-checker should verify the accuracy of quotations taken from field interviews. (We recommend that teams do so.) In these cases, authors should provide the relevant page(s) of interview transcripts to the fact-checker.

Oftentimes, authors will summarize an interview or published source rather than provide direct quotes. For example, an author might write, "During the meeting, Smith and Jones expressed a deep concern over the health risks of smoking," or "The health risks associated with smoking are well documented in the literature (Smith, 1975; Jones, 1980)." In such instances, we suggest that the author be held responsible for making sure that the positions are represented accurately. Unless requested by the author or the report team, the fact-checker would *not* conduct an independent review the field notes or publications to verify accuracy. If the text in question contained a published source, the fact-checker should verify that the names and years are accurate and that the citation is included in the reference section.

- **Legal or policy descriptions**

Fact-checkers should confirm that all legal or policy descriptions are completely accurate (including, for example, the particulars of the policy itself, the date that the law was enacted, etc.). The author should provide copies of source information (legislation, policy manuals, etc.) for the fact-checker to use for verification purposes. If there are no written sources, the fact-checker should ask the author whether there is an "in-house expert" who may check policy descriptions. For example, a team may identify a staff member who has extensive knowledge of Los Angeles County's welfare plan to verify information on the county's policies.

In some cases, state officials, site contacts, or other experts outside of MDRC may be asked to verify the accuracy of a policy description. We would normally do this only when no published sources were available. If asking site contacts to fact-check, it may be most efficient to ask them to do so when they are reviewing the draft document. A cover letter could ask them to focus on the accuracy of specific paragraphs or sections. Whenever someone from outside of MDRC checks a description, the fact-checker should document that person's name, position, and the date next to the verified information.

- **Tables and figures**

The kinds of qualitative data presented above can appear in either text or tables and figures. Fact-checkers should scrutinize carefully all tables and figures because the information they contain is more likely to be looked at by readers and pulled out from the rest of the report. As with quantitative tables, fact-checkers should check the information contained in qualitative tables and figures first, prior to checking any text. Once this is done, all references to and discussion of the tables and figures in the text can be checked against the tables and figures as opposed to the original source. Each time any information in a table changes, the information in both the table and text must be fact-checked again. The type of qualitative data in tables and figures should be checked according to the guidelines presented above.

2. Verification Involving Report Reviewers and Authors

- **Summary and concluding statements**

We suggest that authors and report reviewers are primarily responsible for fact-checking summaries and conclusions. Reviewers' main focus should be on whether the summary and concluding statements are adequately supported by the evidence presented in the chapter. In addition, they should consider whether the statements are clear and reflect an appropriate tone. Authors should also critically examine their summary and concluding statements. Summary and concluding statements that contain any of the types of facts covered above (e.g., statements that use quantitative terms or modifiers, direct quotes, etc.) should be verified according to the procedures we have outlined.

3. Verification Solely by Authors

- **Program operating descriptions**

Descriptions of program operating procedures are central to most implementation chapters and reports and often help us draw major conclusions (e.g., whether a program model received a fair test, whether services were delivered as policy makers intended, etc.). Because these descriptions are typically culled from many pages of field notes (and sometimes the author's memory), it makes little sense — and may be impractical — to ask fact-checkers to review the source material. Thus, we recommend that authors take primary responsibility for verifying the information themselves.

To protect against errors, authors should work closely with other team members to make sure that program descriptions are accurate. Ideally, at least one other team member will have visited the program offices or will be knowledgeable about the program. (This relates to how we do field research and may be addressed in a future working group.) Authors may also ask an "in-house" expert or site contact to check program descriptions. Authors should always keep track of the information they used (e.g., specified sets of field notes, program procedures manuals, etc.) so that any questions about their source materials or interpretations can be quickly resolved.

- **Vignettes and profiles**

Vignettes and profiles are intended mainly for illustration purposes and are lower priority for fact-checking. Indeed, some information may be changed intentionally to protect confidentiality. Since vignettes and profiles often represent a composite of interview and observational data, we suggest that authors be held responsible for getting the key points right. If an author or report team desires an independent review by a fact-checker, the author should provide the fact-checker with the relevant sections of interviews or field notes. The fact-checker should use reasonable judgment to assess whether or not the vignette accurately depicts the scenario as described by the author's source material.

- **Statements of general knowledge**

Statements of general knowledge are difficult or impossible for a fact-checker to verify, so we suggest that authors be held responsible for getting these right. If an author is uncertain about a statement, s/he may want to run the statement by a co-worker for verification.

Verifying Internal Consistency

As noted earlier, *internal consistency* refers to the need for a report to describe the same thing (e.g. a program, policy, finding, etc.) in the same way (e.g. strict, lenient, significant, insignificant, etc.) and to use the same language to refer to the same concepts (e.g. The New Hope Project versus The New Hope Program versus New Hope). Two types of internal consistency should be checked in all reports: (1) consistency throughout a report (including references to other chapters) and (2) consistency within a chapter. Checking internal consistency is a primary responsibility of the fact-checker, although report coordinators, authors and reviewers also play a role.

- Report coordinators, working with other team members, should compile a list of terminology used in the report or project (sometimes editors compile these lists). At a team's discretion, they may also develop standards for using quantitative terms or modifiers (i.e., what constitutes most, large, etc.).

- Fact-checkers (or the report coordinator) should check terms against the terminology list, check that quantitative terms or modifiers are used consistently, and consult each other to check references to other chapters. For example, if the person checking Chapter 3 finds a reference to Chapter 1, that person should ask the person checking Chapter 1 if the statement in Chapter 3 is correct.

- Report authors should review introduction, overview, and conclusion chapters for agreement with their own chapters and accuracy of information in their areas of expertise.

- Report reviewers should be asked to pay close attention to consistency across chapters, particularly in regards to the characterization of the findings.

Attachment A: Commonly Used Quantitative Terms or Modifiers

Quantifying statements or words	
all	overall
consistently	about half
virtually all	majority
some	couple
many	seemed
common	typically
other	especially (group)
none/ no one	few
few	at least (#)
lack of	biggest (concern, factor)
little	main (concern, factor)
several	recurring (concern, factor)
a number of	in particular
most	overwhelming
whenever respondents talked about…, they said	in general, generally
recipients/respondents/ethnographers/they/women (implying all, but without a quantifier)	as a group
Comparisons	
more/less than (this group more than that group)	
implied comparisons (e.g., About the economy, one group expressed fear…[next paragraph] Another group didn't mention the economy as a factor in their job search…)	

Attachment B: Hints and Tips for Authors and Consultants

Authors and consultants should keep in mind throughout data collection, data analysis, and report writing that they will eventually be held accountable for many of the "facts" that they produce or write. Following are some hints and tips to facilitate this:

⇒ **Keep everything**. Keep all documentation materials including field notes, e-mail, notes of calculations, etc. As you go along, collect all these materials in a folder or binder, so when you need to pass them on to a fact-checker, they are already centralized.

⇒ **Document sources as you write**. We think it is more efficient to note data sources in the text as you write. When you have the source material in front of you, it just takes a little time to note, for example, that a specific paragraph in your chapter summarizes information from pp. 2-4 of the 1998 field notes. If you wait until the fact-checking phase of the report process, which usually occurs months after a first draft is written, it may be difficult and time-consuming to reconstruct what you did. Authors can indicate sources in the text using endnotes or bracketed text; after fact-checking, these can be edited out.

⇒ **Use the annotation function in Word.** Sources can also be noted using annotations. The advantages of using annotations rather than endnotes or brackets are that the source material can either be hidden — i.e., not printed with the text — or shown at your discretion, and since it can be hidden, you never need to delete it from the document.

This function creates an endnote-type annotation in the text in which the author can write the source of the fact (for example, field notes, 3/99, p. 17-20; Table 2.2, Hamilton and Brock, 1994). Annotations can also be used to indicate the background or evidence for a statement that uses a quantitative term or modifier. For example, if you write: "Most case managers felt…" you could use an annotation to note: "16 of 20 case managers, from field notes, 3/99."

To create an annotation, use your mouse and click on *Insert*, and then *Annotation*. This will create an endnote-type annotation using your initials; it looks like this in the text: [ses1] (the second annotation would be [ses2], etc.) You can make the annotation numbers appear or disappear on your screen by clicking the *Show/Hide button*, which looks like this: ¶ .

To print the annotations, click on *Tools*, and then *Options*. Click on the *Print* tab, and under the heading "Include with Document," select *Annotations*. This will print the annotation numbers in the text where you inserted them, and will print the annotations at the end of the document.

⇒ **Use the tools that NUD•IST provides.** When NUD•IST is used to help analyze qualitative data, it can be very helpful in the fact-checking process. For example, when you use information from a certain part of a document, note the line numbers. If you use information from reports that you ran, note and save the reports that were used, and include them in the fact-checking packet you give to the fact-checker.

⇒ **Keep an in-progress list of references**. Keep a list of the published sources that you use, updating it as sources are added or removed, so you can easily compile whatever you need for fact-checking. Make a copy of the title page of each reference you use, and include these copies in the fact-checking packet you give to the fact-checker. This will facilitate both finalizing the citations and compiling the final reference list for the report.

Bibliography

Becker, Howard S. 1998. *Tricks of the Trade: How to Think about Your Research While You're Doing It*. Chicago: University of Chicago Press.

Brock, Thomas, Laura C. Nelson, Megan Reiter. 2002. *Readying Welfare Recipients for Work: Lessons from Four Big Cities as They Implement Welfare Reform*. New York: Manpower Demonstration Research Corporation.

Creswell, John W. and Raymond C. Maietta. (forthcoming). "Systematic Procedures of Inquiry and Computer Data Analysis Software for Qualitative Research," in N.J. Salkind, Ed., *Handbook of Research Design and Social Measurement*, 6th Edition.

Denzin, Norman K. and Yvonna S. Lincoln, Eds. 1998. *Strategies of Qualitative Inquiry*. London: Sage Publications.

Emerson, Robert M., Rachel I. Fretz, Linda L. Shaw. 1995. *Writing Ethnographic Fieldnotes*. Chicago: University of Chicago Press.

Ford, Kate, Iddo Oberski, Steve Higgins. 2000. "Computer-Aided qualitative Analysis of Interview Data: Some Recommendations for Collaborative Working," *The Qualitative Report*, 4(3-4): . (http://www.nova.edu/ssss/QR/QR4-3/oberski.html)

Froggatt, K.A. 2001. "The analysis of qualitative data: processes and pitfalls." *Palliative Medicine*, vol. 15, no. 5, pp. 433-438.

Glaser, Barney G. and Anselm L. Strauss. 1967. *Discovery of Grounded Theory: Strategies for Qualitative Research*. New York: Aldine de Gruyter.

Kato, Linda Y. and James A. Riccio. 2001. *Building New Partnerships for Employment: Collaboration Among Agencies and Public Housing Residents in the Jobs-Plus Demonstration*. New York: Manpower Demonstration Research Corporation.

LeCompte, Margaret D. 2000. "Analyzing Qualitative Data." *Theory into Practice*; vol. 39 no.3 pp.146-54.

LeCompte, Margaret D. and Jean J. Schensul. 1999. *Ethnographer's Toolkit 5: Analyzing and Interpreting Ethnographic Data*. Walnut Creek, CA: Altamira Press.

Li, Mao-Neng Fred and Gary J. Lautenschlager. 1999. "IASGA: A SAS MACRO Program for Interrater Agreement Studies of Qualitative Data via a Generalizability Approach." *Educational and Psychological Measurement*, vol. 59 no. 3 pp. 532-37.

Lofland, John and Lyn H. Lofland. 1995. *Analyzing Social Settings: A Guide to Qualitative Observation and Analysis*. New York: Wadsworth Publishing.

Matus-Grossman, Lisa and Susan Gooden. 2002. *Opening Doors: Students' Perspectives on Juggling Work, Family, and College*. New York: Manpower Demonstration Research Corporation.

Merriam, Sharon B. 2001. *Qualitative Research and Case Study Applications in Education*. San Francisco: Jossey-Bass.

Miles, Matthew B. and A. Michael Huberman. 1994. *Qualitative Data Analysis: An Expanded Sourcebook* (2nd Edition). London: Sage Publications.

Morison M. and J.Moir. 1998. "The role of computer software in the analysis of qualitative data: efficient clerk, research assistant or Trojan horse?" *Journal of Advanced Nursing*, vol. 28, no. 1, pp. 106-116(11).

Polit, Denise F., Rebecca Widom, Kathryn Edin, Stan Bowie, Andrew S. London, Ellen K. Scott, Abel Valenzuela. 2001. *Is Work Enough? The Experiences of Current and Former Welfare Mothers Who Work*. New York: Manpower Demonstration Research Corporation.

Richards, Lyn. 2000. *Using N5 in Qualitative Research*. Bundoora, Victoria: QSR International,.

Richards, Tom. 2000. *N5 Reference Guide*. Bundoora, Victoria: QSR International.

Richards, T. and L. Richards. 1995. Using hierarchical categories in qualitative data analysis. In U. Kelle (Ed.), *Computer-Aided Qualitative Analysis*: *Theory, Methods and Practice*. London: Sage Publications.

Seidell, John and Udo Kelle. 1995. "Different Functions of Coding in the Analysis of Textual Data," in Udo Kelle, Ed., *Computer-Aided Qualitative Data Analysis: Theory, Methods and Practice*. London: Sage Publications.

Smith, C. and Short P.M. 2001. "Integrating Technology to Improve the Efficiency of Qualitative Data Analysis-A Note on Methods," *Qualitative Sociology*, vol. 24, no. 3, pp. 401-407(7).

Szabo, Vivian and Vicki R. Strang. 1997. "Secondary Analysis of Qualitative Data." *Advances in Nursing Science*, vol. 20, no. 2, p. 66.

Walberg, Herbert J., Richard Niemiec, Herbet J. Walberg III. 1997. "A Technique for Objective Analysis of Qualitative Data." *Evaluation Practice*, vol. 18, no. 1, p. 55.

Willis, Jerry and Muktha Jost. 1999. "Software for Analyzing Qualitative Data." *Computers in the Schools*, vol. 15 no. 3-4 pp. 117-50.