

## **DRAFT**

### **TIER Protocol (version 2.1): PROCESS**

Constructing replication documentation should be an integral part of your workflow throughout the entire research process—not something you do as an afterthought when you have finished your paper.

You should begin constructing your documentation before you even start working with your data, and add to it incrementally as your research progresses.

By the time you have finished the project and written your paper, only a few final additions and checks should be required to complete the documentation.

# Think of your research in terms of three major phases:

## Pre-data: Before you start working with your data

There are two tasks to complete before you start working with your data:

### *Pre-data 1: Construct a hierarchy of empty folders.*

You will use these folders to save your work while you conduct your research, and when you are finished you will store your complete replication documentation in them.

### *Pre-data 2: Create three blank documents.*

You will record information in these documents as your work progresses, and at the end they will be part of your replication documentation.

## Data Work: Working with your data

The work you do with your data can be broken down into the following steps:

### *Data Work 1: Documenting your original data.*

Each time you obtain a new file containing data you will use for your project, you should preserve a copy to include in your replication documentation, and record some information about it in one of the blank documents you created.

### *Data Work 2: Writing command files that process your data.*

In these command files, you write code that transforms the data files with which you begin your project into the data files you use to generate the results for your paper.

### *Data Work 3: Constructing a Data Appendix.*

This document serves as a codebook for your analysis data files.

### *Data Work 4: Writing command files that generate your results.*

In these command files, you write code that executes the procedures or analyses that generate the results you report in your paper.

## **Wrap-up: Wrapping things up**

After you finish the analysis and write your paper, a few steps remain to complete the replication documentation:

### *Wrap-up 1: Writing a Read Me file.*

As described above, you create a blank Read Me file at the very beginning of the project. As described later in this document, you will record some information in the Read Me file at various points in your work. Other information that should be in the Read Me file can only be entered after you have completed your paper.

### *Wrap-up 2: Proofreading and testing your command files.*

Proofread your command files to be sure that they are concise and that you have provided sufficient comments, and test them to be sure that they run without errors and successfully replicate your study.

### *Wrap-up 3: Making a final check of all your replication documentation.*

Check that your replication documentation contains all the components described in the [specifications of the TIER Protocol](#), and that they are stored in the hierarchy of folders and sub-folders called for by the Protocol.

## **Pre-Data 1: Construct a hierarchy of empty folders.**

Begin by making a hierarchy of empty folders and sub-folders to store your work in.

You will use these folders to save data files, command files and other documents as you assemble and construct them throughout the course of your work on the project.

When you have finished, you will store the replication documentation for the completed project in these folders.

### **Choose a safe place to save your work.**

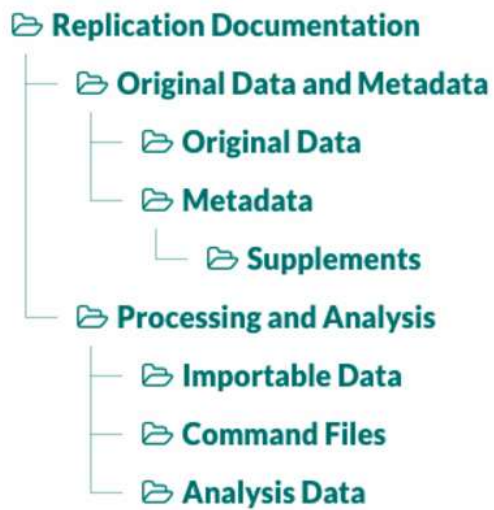
Decide where you will keep the various files you will be working with over the course of your project—such as a dedicated folder on the hard-disk of your computer, a server maintained by your institution, or a web-based platform like Dropbox, GitHub or the Open Science Framework (OSF).

Choose a place that is secure and stable, and easily accessible to everyone who will be working on the project.

Decide on a reliable system for backing up your work.

### **Create the hierarchy of folders and sub-folders specified by the TIER Protocol.**

The hierarchy of folders and sub-folders is described in detail in the [specifications of the TIER Protocol](#), and illustrated below.



**Save the folders you created in the safe place you chose to save your work.**

Throughout your work on the project, you will be adding new documents to these folders and editing ones that you have already stored there.

## **Pre-Data 2: Create three blank documents.**

Using whatever word-processing or text-editing software you choose, create three blank documents that will eventually become part of your replication documentation.

You will add information to these documents throughout the process of conducting your research.

The three documents to create are:

A Read Me file

A Metadata Guide

A Data Appendix

### **Type an appropriate title at the beginning of each document.**

These hypothetical examples illustrate the kinds of titles you might choose:

Read Me File for Econometrics Project by A. Smith and B. Jones.

Metadata Guide for J. Student Senior Thesis.

Data Appendix for “The Economics and Politics of Popular Stringed Instruments,” by U. K. Laylee.

Other than their titles, leave these documents blank.

### **Give these documents appropriate names, and save them in the appropriate folders.**

Give your Read Me file the name *ReadMe.EXT*, and save it in the top level of your Replication Documentation folder.

Give your Metadata Guide the name *MetadataGuide.EXT*, and save it in the top level of your Metadata folder.

Give your Data Appendix the name *DataAppendix.EXT*, and save it your Analysis Data folder.

**Note:** In the above file names, *.EXT* represents the extension used on the names of documents created with your word-processing or text-editing software. (For example, if you are using Microsoft Word, *.EXT* would be replaced by *.docx*.)

## **Data Work 1: Documenting your original data.**

Any document from which you extract statistical data to use in your project is called an *original data file*.

In some cases, all of the data for a project may come from a single original data file; in others, the data are extracted from two or more original data files.

Each time you obtain an original data file, there are several things you should do immediately:

### **Save a copy of the original data file in your Original Data folder.**

You may give the file a new name when you save it in the Original Data folder, but other than that the copy you save should be identical to the original version of the data file. The contents and format should not be modified in any way.

### **Enter information about the original data file in your Metadata Guide.**

The Metadata Guide is one of the three documents you created before you started working with your data.

Each time you obtain a new original data file, add a section about that file to the Metadata Guide.

Begin the section with a header that identifies the original data file it pertains to (e.g., “Metadata for penn\_tables\_1986\_2010.txt”).

Review the specifications for the Metadata Guide given in the TIER Protocol, then enter the relevant information about the original data file in your Metadata Guide.

If you enter the required information each time you obtain a new original data file, your Metadata Guide will be complete well before you finish your project.



## **Save an exact or modified copy of the original data file in your Importable Data folder.**

For every original data file you store in your Original Data folder, you should store one corresponding importable data file in your Importable Data folder.

*If the original data file is in a format that can be opened by the statistical software you are using for the project*, the corresponding importable data file should be an exact copy of the original. In this case, the names of the original and importable data files should be the same.

*If the original data file is in a format your software cannot open*, the importable version should be modified to the *minimal extent necessary* to make it possible for your software to open it. In this case, the names of the original and importable data files should be different.

## **Write a note in your Read Me file explaining any modifications you made to the original data file when you created the importable version.**

The Read Me file is one of the three documents you created before you started working with your data.

Each time you need to modify an original data file to create the importable version, you should make a note of the changes you made in the Read Me file.

By the time you have finished the project, you will have made a note in your Read Me file about every original data file you had to modify to create an importable version.

You do not need to make a note of this kind when the importable version of a data file is identical to the original.

You should not begin working with the data until you have completed all of these steps.

## Data Work 2: Processing your data.

Conducting all the steps necessary to transform your [importable data files](#) into one or more [analysis data files](#) that you will use to generate the results you report in your paper.

Processing your data to prepare them for the analyses you will report in your paper.

The processing phase of a project consists of all the steps involved in transforming your [importable data files](#) into the [analysis data files](#) that you use to generate your results.

**In one or more command files, write code that transforms your importable data files into your analysis data files.**

Exactly what steps of processing are required varies a great deal, but examples of some common procedures include:

- Having your software open your [importable data files](#).
- Cleaning the data to resolve any errors or discrepancies.
- Removing variables or cases that you do not need.
- Combining data from different importable data files.
- Transposing a data table so that columns become rows and rows become columns.
- Generating new variables.
- Saving your analysis data files.

## **Decide how to organize the code that processes your data into one or more command files.**

It is always possible to put all the necessary commands in a single file.

But in many cases separating different parts of the processing phase into different command files can help you keep track of what you are doing.

The best way of dividing your data processing among command files will depend on the particulars of your project, but the following scheme often works well.

- For every importable data file you have, write one command file that reads the data it contains, cleans the data as necessary to prepare them for merging with the data from the other importable data files, and then saves them in a new file, in the native format of the software you are using.
- Then write one additional command file that merges all these natively formatted files, processes them as necessary to construct the analysis data files, and then saves the analysis data files in your software's native format.
- Depending on the number of importable data files you have and how the data in them are organized, other schemes for dividing the processing phase among your command files may be more convenient; you should use whatever scheme you find works best for your project.

Whatever scheme you choose, you will explain in your [Read Me file](#) the order in which the command files need to be run to replicate your project.

## **Save your command files and your analysis data files in the appropriate folders.**

Save the command files that process your data and create your analysis data file(s) in your [Command Files folder](#).

Save the analysis data file(s) in you [Analysis Data folder](#).

## Data Work 3: Constructing your Data Appendix.

The [Data Appendix](#) provides information about all the variables in your [analysis data files](#), such as their names, definitions, coding, and summary statistics.

The Data Appendix serves as a codebook and users' guide for your analysis data files.

The Data Appendix is one of the three documents you created before you began working with your data.

You should construct your Data Appendix as soon as you have finished writing the command files that create your [analysis data files](#).

When you construct your Data Appendix, you are likely to learn things about your data that you should know before you begin your analysis. So you should not begin the analysis until after you have constructed your Data Appendix.

### **Review the specifications for the information that should be included in the Data Appendix.**

The TIER Protocol gives complete specifications of the information that should be included in the Data Appendix.

To summarize the specifications briefly: The Data Appendix should provide information about every variable in your [analysis data files](#), including definitions and coding (for all variables), summary statistics and histograms (for quantitative variables), and relative frequency tables and charts (for categorical variables).

**Write a command file that generates all the descriptive statistics, tables and figures needed for the Data Appendix.**

These descriptive statistics, tables and figures should be created using the data in your [analysis data files](#).

Give this command file the name *DataAppendix.CMD*, where *CMD* represents the extension your statistical software uses for the names of command files.

Save *DataAppendix.CMD* in your Analysis Data folder.

**Finish composing the Data Appendix, inserting the descriptive statistics, tables and figures in the appropriate places.**

When you have finished, save the Data Appendix in your [Analysis Data folder](#).

## Data Work 4: Analyzing your data.

Performing the procedures on your [analysis data files](#) that generate the figures, tables and other statistical results you report in your paper.

Performing the analyses and procedures that generate the results you report in your paper.

In the analysis phase, you generate the results of your study.

The results consist of all the findings you report in your paper that are based on computations performed on your [analysis data files](#). They may be presented in various forms, including tables, figures, and numerical values reported in the text of the paper.

**In one or more command files, write code that generates all the results you report in your paper.**

These command files should contain commands that open up your [analysis data files](#), and then use those data to generate the output upon which your [results](#) are based.

Every command that generates any of your results should be preceded by a comment that states which result the command generates. A few Hypothetical examples illustrate what these comments might look like:

\* The following command generates the first column of Table 6.

\* The following command generates the second column of Table 6.

\* The following command generates Figure 4.

/\* The following command generates the correlation of 0.31 between the variables *INC* (individual annual income, reported in the natural log of current US dollars) and *SATIS* (individual subjective self-report of overall satisfaction with life, on a scale of 0—least satisfied—to 10—most satisfied). This correlation is reported on page 27 of the paper. \*/

The command files for your analysis phase should not contain any commands that generate new variables or process your data in any way. All the procedures required to prepare your data for analysis should be executed by the command files you wrote for the processing phase.

It is often convenient to write all the commands for the analysis phase in a single command file. However, if the nature of your project or the structure of your data are such that you think it would make sense to divide the code that generates the results into two or more command files, you should feel free to do so. No matter how you organize your analysis command files, your [Read Me file](#) will include an explanation of how to use them to reproduce your results.

Save the command files you write for the analysis phase in the [Command Files folder](#).

## Wrap-up 1: Finishing your Read Me file

If you follow the research process described in this document, you construct your replication documentation incrementally throughout the course of your work on the project. By the time you have completed your final paper, your replication documentation should also be nearly complete.

The Read Me file is one of the three documents you created before you began working with your data.

Review the specifications for the information that should be included in the Read Me file given in the specifications of the TIER Protocol.

You should already have recorded one part of the required information, namely **notes explaining any modifications you made to the original data** files when you made importable versions of them.

To finish your Read Me file, you should add the other items specified by the TIER Protocol:

- An overview of all the files included in the replication documentation, and the structure of the folders in which they are stored.
- Step-by-step instructions for using the replication documentation to replicate the study.



## **Wrap-up 2: Proofreading and testing your command files.**

In the course of your project, you constructed command files that processed your data, produced the descriptive statistics and figures for your Data Appendix, and executed the analyses and procedures that generated the results you reported in your paper.

But before you consider your command files to be complete and ready to store in your final replication documentation, you should edit them for accuracy and clarity, and test them to be sure they reproduce the results of your project as intended.

### **Editing Your Command Files**

Edit all your command files to be sure they are accurate, concise, and free of detritus.

- Remove any commands that turned out not to be necessary for your project.
- If you realize in hindsight that the code you wrote to execute any of the procedures could be rewritten in a simpler or more streamlined way, then revise the code accordingly.
- Be sure the comments in your command files are extensive and clear enough to allow someone else to understand what is accomplished in each step of data processing and analysis.

### **Testing Your Command Files**

Test your command files to be sure that they all run without error and that they successfully reproduce the results you reported in your paper.

- Try following the instructions for replicating your project that you wrote in the Read Me file to be sure that all your command files run without a hitch and produce the intended output.
- If you encounter any errors or crashes, diagnose and fix the problem, and then start the test over.

## Wrap-up 3: Making a final check of all your replication documentation.

Before you consider your replication documentation to be complete and final, check to be sure that it satisfies all the requirements of the TIER Documentation Protocol.

First, review the specifications of the [TIER Documentation Protocol](#).

Then check that:

- All the required files are included in your replication documentation, and that they are stored in the correct folders.
- The content and format of every file meet the specifications of the TIER Protocol.

Finally, delete any extraneous files.

- Your replication documentation should contain only the files specified by the TIER Protocol, or that you intentionally chose to include for a particular purpose.

**After you check that your replication documentation contains everything that it should, and that it does not contain anything extraneous, you may consider it complete.**

