i2i DIME
TRANSFORM DEVELOPMENT

Research Reproducibility: Folder Structure and Master Scripts

Roshni Khincha

DIME Analytics – World Bank

November 11, 2019

Overview

Overview

Think reproducible

The applicable goal of DIME Research Standards for this session is to ensure that all the research we develop is reproducible

Publishing a paper by itself is not enough anymore

- Like tables and figures, code is now an equally important output to share In this context, publishing code is pointless if it is
  - not reproducible, (i.e. not getting the same results) and
  - no one understands how to run it
- For code to be useful, it requires transparency, accountability, and an easy to understand workflow
- Basically, code needs to be organized and readable

This is a lot easier said than done

- At DIME, we have large teams collaborating on the same codes and data sets
- Long projects easily become complex as they have multiple rounds/sources of data which need to be organized
- Standardizing organization of documents and code prevents mistakes and reduces the cost of transitioning across projects and teams

What are we doing in the next few hours?

In this session we will understand and implement best practices to manage data work through

- Setting up a good folder structure
- Setting up a master script which runs all code

The goal for this session is that anyone with full access to your project files and folders should be able to replicate the research understands which file(s) need to be run in which order

Overview

Thinking time! - The big picture

Now let's take some time to think through the big picture for your project

- What are all the sources of data your project uses?
  - Primary data sources (i.e. any data collected for the first time for the purpose of your project)
    - Eg: Surveys, Monitoring data, Admin data
  - Secondary data sources (i.e. any data collected by someone else but you are using for your project)
    - Eg: Census data, Admin data, Web-scraped data, Public APIs data
  - Make a list of these on the paper provided

Overview

DataWork Folder: Overview

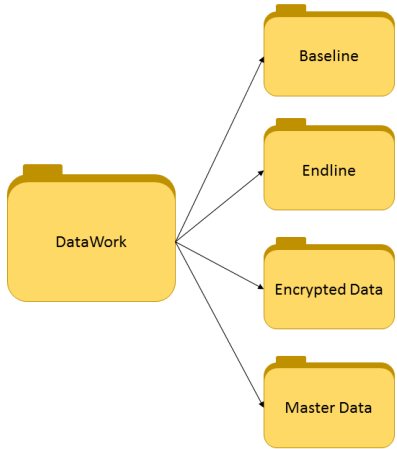
- Now that we have a sense of all the data on our projects, let's understand how to set up an organized and easy to understand folder structure
- *iefolder* provides a template for the folder structure for a typical DIME project
- However, more and more projects are not typical DIME projects
- Regardless, the templates used in *iefolder* is a great starting point to think through the folder structure for your project

DataWork Folder: Overview

- Your project folder probably has a lot of sub folders for literature, presentations, concept notes, and other documents
- We will focus on the data folders in your project which we will store in a DataWork sub-folder
- This sub-folder will have a standardized structure so it's easy for everyone to find things in it and move across projects
- We will run through a set up of an efficient folder structure for a data with multiple rounds of primary data
- Note that, the set up for each project will be unique to the project but we are just running through a template here

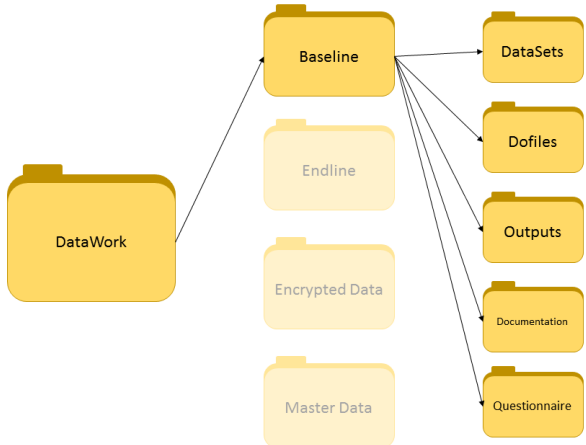
DataWork Folder: Overview

This is what it will look like:



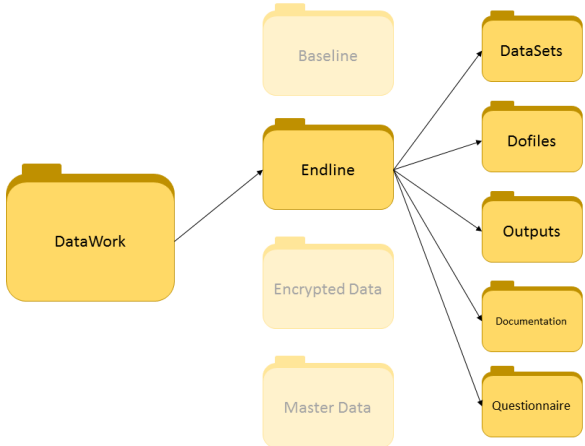
DataWork Folder: Rounds folders

The baseline folder will store all baseline data, as well as do-files and outputs that refer exclusively to this round of data collection.



DataWork Folder: Rounds folders

The endline folder will store all endline data, as well as do-files and outputs that refer exclusively to this round of data collection. Note that its sctructure is exactly the same as the structure of the baseline folder.

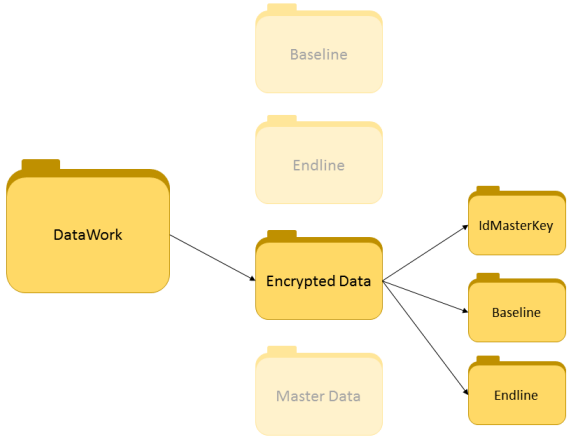


DataWork Folder: Rounds folders

- You can create as many rounds folders as you want
- That is, every round of data collection from your project should have its own round folder

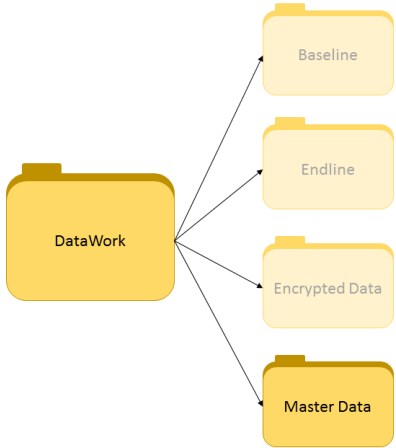
DataWork Folder: Encrypted Data

The encrypted data folder will contain all personally identifiable data for each round of data collection, and a folder with ID master keys linking each unidentified ID to the identified observations. As the name suggests, this folder should be encrypted.



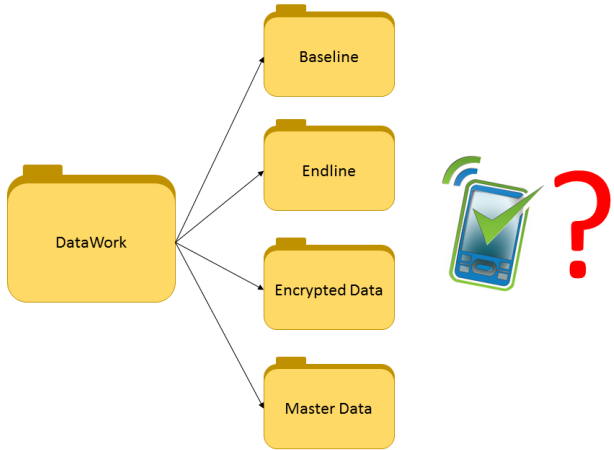
DataWork Folder: Master Data

The master data folder will store the master data sets for each unit of observation in your project.

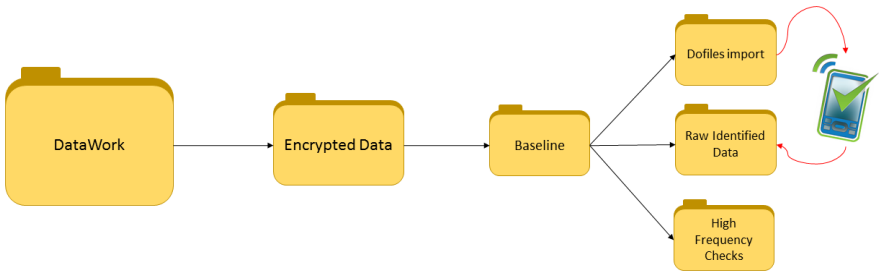


Using the DataWork Folder

So you received data from the field. What now?

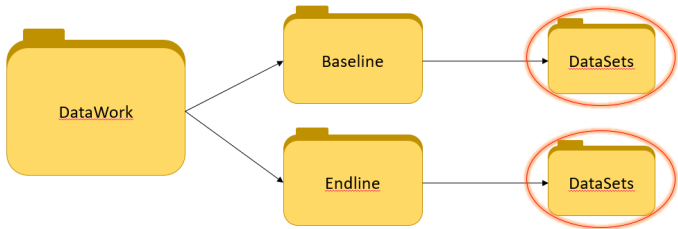


Using the DataWork Folder: EncryptedData



- The raw data with identifying information should be stored in the EncryptedData folder
- The do-files used to import your data from SurveyCTO will also go in this folder

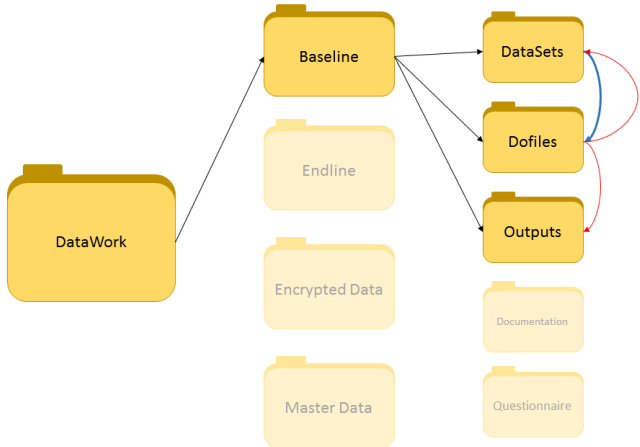
Using the DataWork Folder: DataSets



- All data in the rounds folders should be de-identified
- *Note: We will cover de-identification in a session later in the bootcamp*

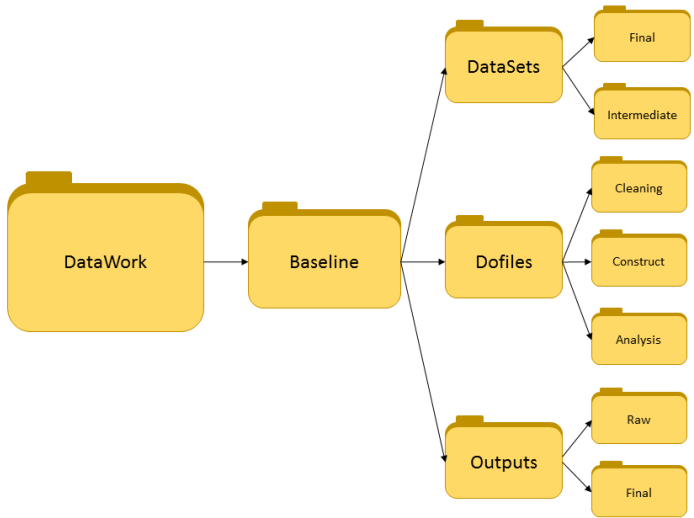
Using the DataWork Folder: Round folders

The do-files in each round folder will load data from that round's DataSets folder and store any outputs in the round's Outputs folder



Using the DataWork Folder: Rounds folders

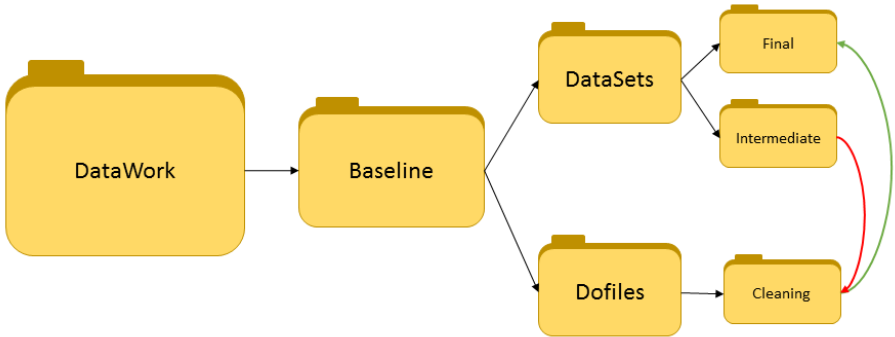
This is what one specific round folder looks like in detail



Using the DataWork Folder: Cleaning

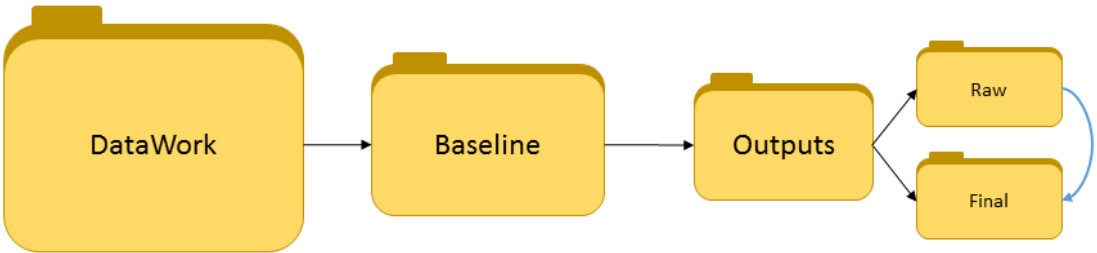
The do-files in the Cleaning Dofiles folder will

1. Load the intermediate data sets (e.g., de-identified survey data)
2. Clean them
3. Save the clean data sets in the Final DataSets folder



Using the DataWork Folder: Outputs

- The Final Outputs folder stores reports, papers and other documents you create using the files in the Raw Outputs folder



Overview

Sketch out your project's folder structure

On the piece of paper, sketch out what you think is the best structure for your project based on your data sources listed.

Remember to think about where each of the following will be stored

- Datasets (Raw, temporary files, Final)
- Code
- Tables and figures
- Documentation

Overview

What's the need for a master script?

- As you might have noticed, we mentioned the creation of many code scripts in the previous slides
- A big project can become very complex, and scripts need to be run in a certain order to create the right output
- That could mean you'd need to write one extremely long script, or a different document with instructions about in which order to run all the scripts
- **However**, you can make a script that runs other scripts
- This makes it easy for anyone reproducing your code to do it with ease

What is a master script?

- The master script is the map over all data work in your data folder
- It is the table of content for the instructions that you code
- It should be possible to follow all data work in the data folder, from raw data to analysis output, by reading the master script

What is a master script?

Here's an example on Stata

```
/*================================================================

*PART 2. - EXECUTE THE CLEANING MASTER DO-FILE
    - add all region names and codes
    - checks that all HHIDs exist in the master data set

================================================================*/

    do "$do/Cleaning/cleaning_master.do"

/*================================================================

*PART 3. - EXECUTE THE CONSTRUCT MASTER DO-FILE
    - add all region names and codes
    - checks that all HHIDs exist in the master data set

================================================================*/

    do "$do/Construct/construct_master.do"

/*================================================================

*PART 4. - EXECUTE THE PANEL CREATION FILE
    - add all region names and codes
    - checks that all HHIDs exist in the master data set

================================================================*/

    do "$do/PanelCreation/panel_create.do"
```

What is a master script?

Here's an example on R

```
#------------------------------------------------------------------------------#
#### 3. SECTIONS  ####

##### 3.2. RMS checks
if(RMS_CHECKS){
  source(file.path(GITHUB, "Road Monitoring/road_monitoring_checks.R"))
}

if(RUN_LAIS_PROCESSING){
  source(file.path(GITHUB,
                   "LAIS&ALIS/LAIS_processing_DRAFT.R"))
}

if(RUN_BENO_PROCESSING){
  source(file.path(GITHUB,
                   "Beno Cars/BenoCars_cleaning.R"))
}
```

Master script: the map to all data work

At the end of every round of a project

- Anyone should be able to follow and to reproduce all your work from raw data to all outputs with one click in this script, after adding only their root folder path
- By reading the master script, someone external to the project should have a general understanding of what is being done at every step

Master script: allows for easy collaboration

- If we share a project over DropBox or GitHub, all team members have the same folder structure
- A master script allows multiple people to set their own global to the project folder
- This way, anyone sharing the project folder can easily run your scripts

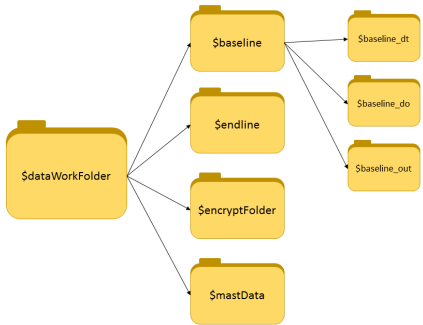
```
*Set this value to the user currently using this file
global user  1   // Luiza
global user  2   // Ben

* Root folder globals
* ---------------------

if $user == 1 {
    global projectfolder "C:\Users\WB501238\Dropbox\DIME\RA survey"
}

if $user == 2 {
    global projectfolder "C:\Users\Benjamin\Dropbox\Work\DIME\RA survey"
}
```

Master script: allows easy updates

The master script contains globals referencing the folders in your DropBox, so you have shortcuts that apply to all folders. Any change in folder structure can be easily accounted for by changing the folder globals

```
    * Project folder globals
    * ----------------------

    global dataWorkFolder         "$projectfolder/DataWork"

*iefolder*1*FolderGlobals*master************************************************
*iefolder will not work properly if the line above is edited

    global mastData               "$dataWorkFolder/MasterData"

*iefolder*1*FolderGlobals*rawData************************************************
*iefolder will not work properly if the line above is edited

    global encryptFolder          "$dataWorkFolder/EncryptedData"
    global masterIdDataSets       "$encryptFolder/IDMasterKey"


*iefolder*1*RoundGlobals*rounds*baseline*****************************************
*iefolder will not work properly if the line above is edited

    *baseline folder globals
    global baseline                "$dataWorkFolder/baseline"
    global baseline_dt             "$baseline/DataSets"
    global baseline_do             "$baseline/Dofiles"
    global baseline_out            "$baseline/Output"
```

(a) Master script



(b) DropBox folder

Overview

Task: Implementing foler structure changes and master script

- Set up the folder structure you drew out on the paper on your Dropbox folder
  - Using *iefolder* to get started is useful even if you are not going to use the exact structure
- Set up a master script template (it will already be done if using *iefolder*c
- Move items into relevant folders
- As you move each item, make sure to update the relevant do files and the master script

Now on to Git and GitHub!