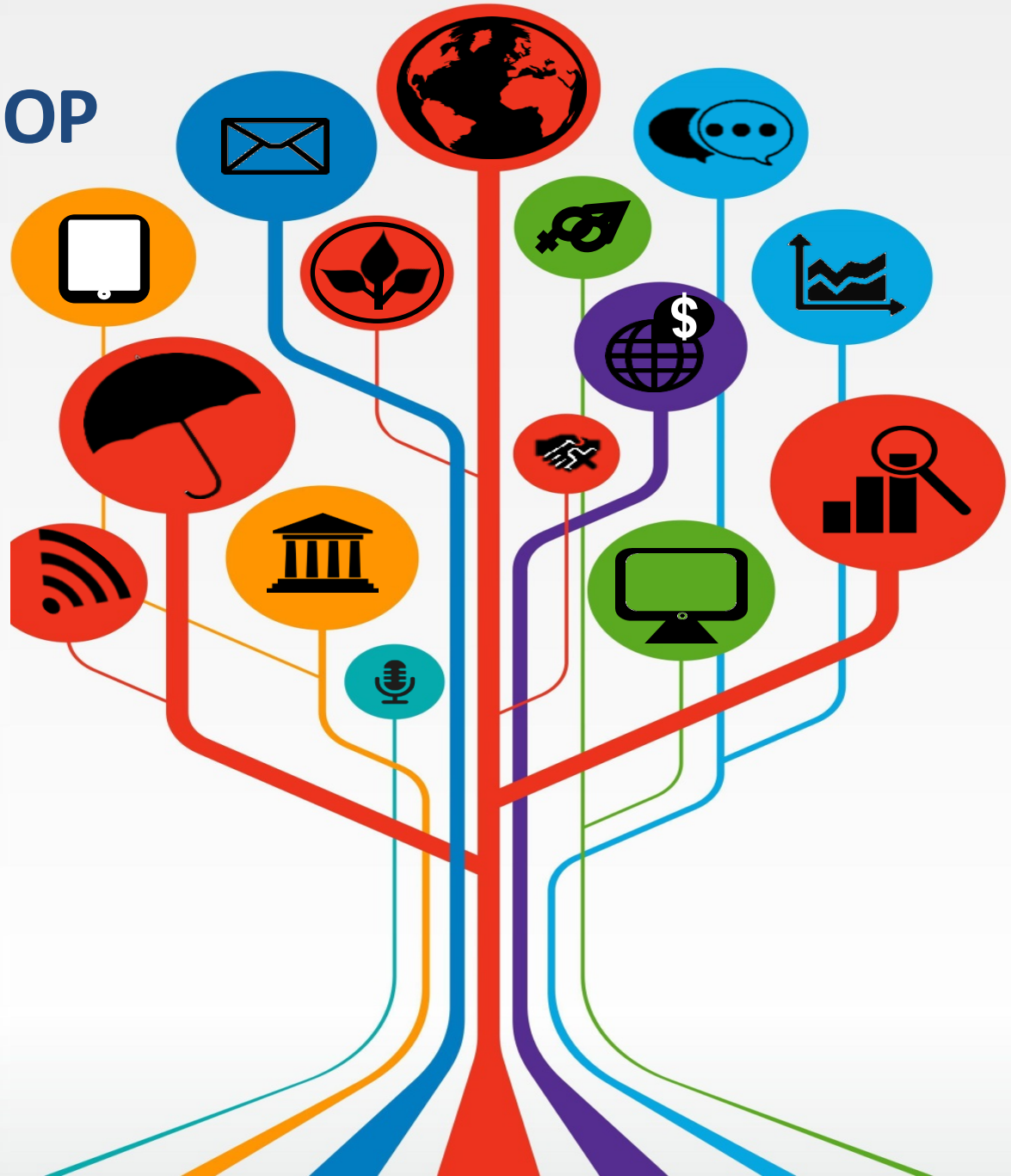
# FIELD COORDINATOR WORKSHOP

## Manage Successful Impact Evaluations

**18 - 22 JUNE 2018**
**WASHINGTON, DC**



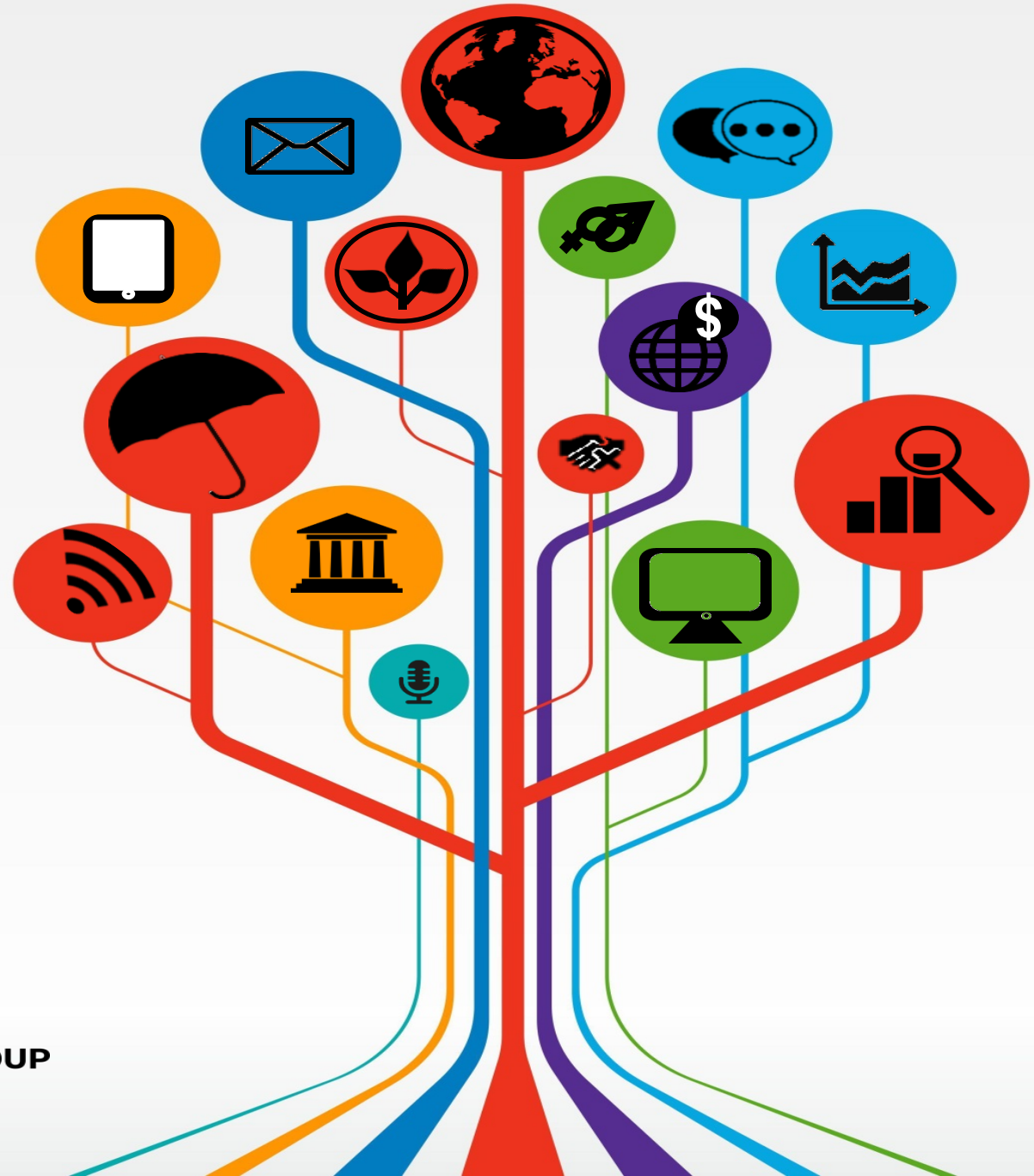
i2i DIME
TRANSFORM DEVELOPMENT

WORLD BANK GROUP

# Real Time Data Quality Checks

Aurelie Rigaud & Kristoffer Bjärkefur

20 June 2018



i2i DIME
TRANSFORM DEVELOPMENT

WORLD BANK GROUP

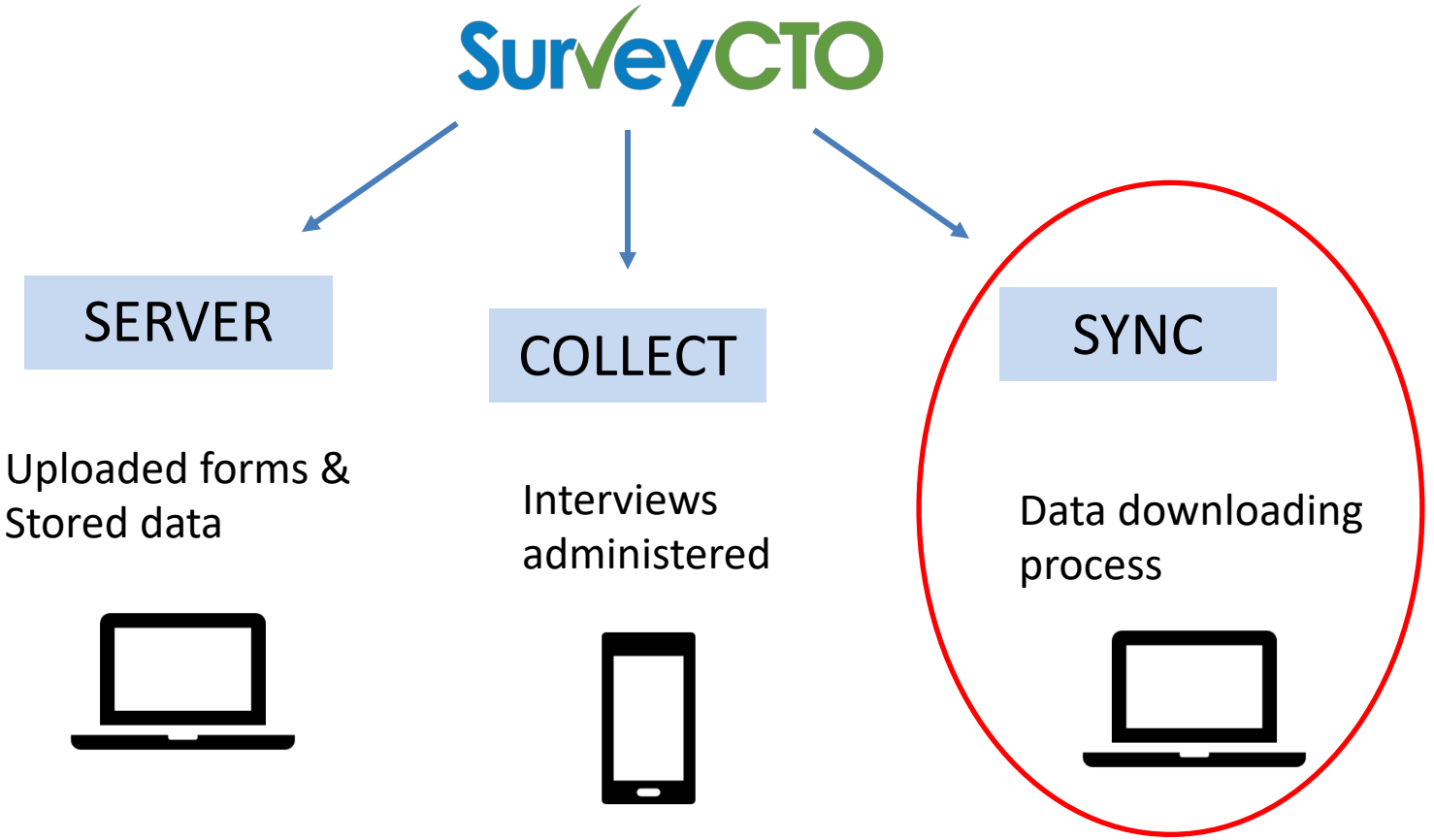
# Objectives

1. Import data
   - Quickly go over how to do this
2. Duplicates
   - Run a prepared file using ‘ieduplicates’ command and understand the output
3. Survey Log
   - Provide the team with clear progress reports
4. Other Manual HFC
   - Run a prepared HFC file and learn to edit it
5. Back checks
   - Set up bcstats, run it and interpret the results

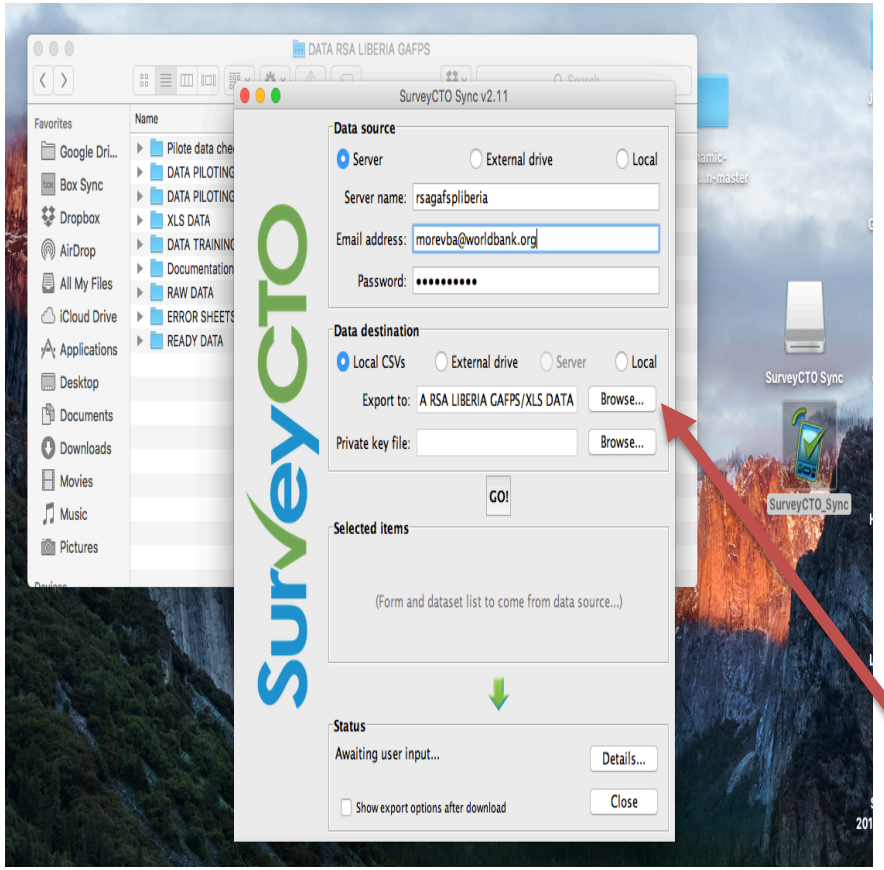
# 1. Import SurveyCTO data to Stata

- **The _WIDE format (Bad practice)**
  - Both from logging in to your server online and through SurveyCTO you can download a file with the suffix _WIDE.
  - Do never use the _WIDE for importing to Stata using the insheet or import command – it is very buggy!
- **Using SurveyCTOs Stata template (good practice)**
  - **Data are download through a special application**
  - In SurveyCTO Sync you can download a Stata do-file together with one .csv file for each *repeat group.*
  - This do-file imports all the .csv in a more correct way
  - It also do a lot of labeling work saving you hours or days of cleaning work

# Import Data

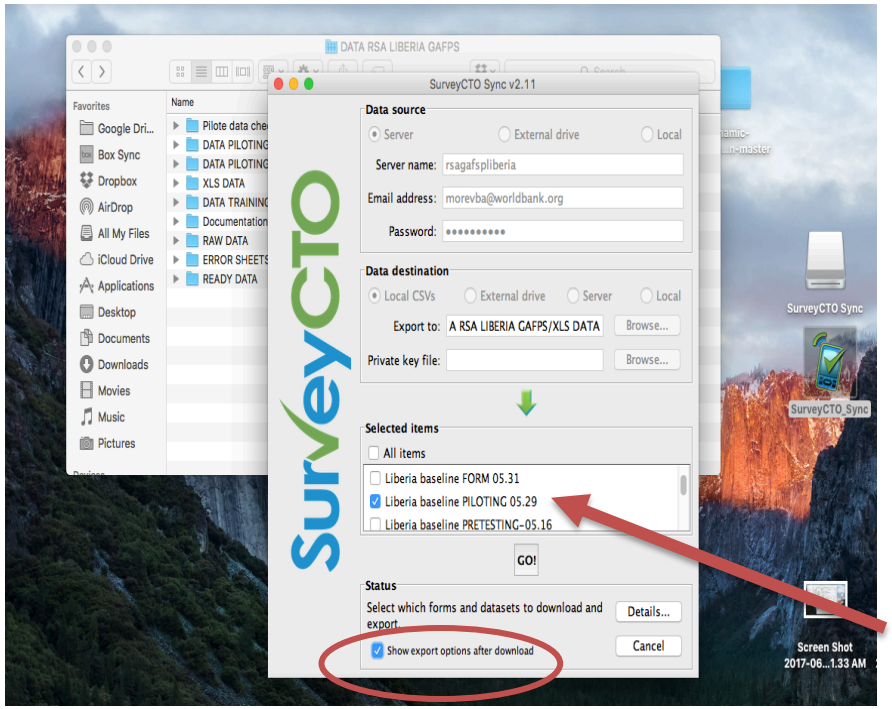

SurveyCTO
SERVER
Uploaded forms &
Stored data
COLLECT
Interviews
administered
SYNC
Data downloading
process

# 1. Import SurveyCTO data to Stata



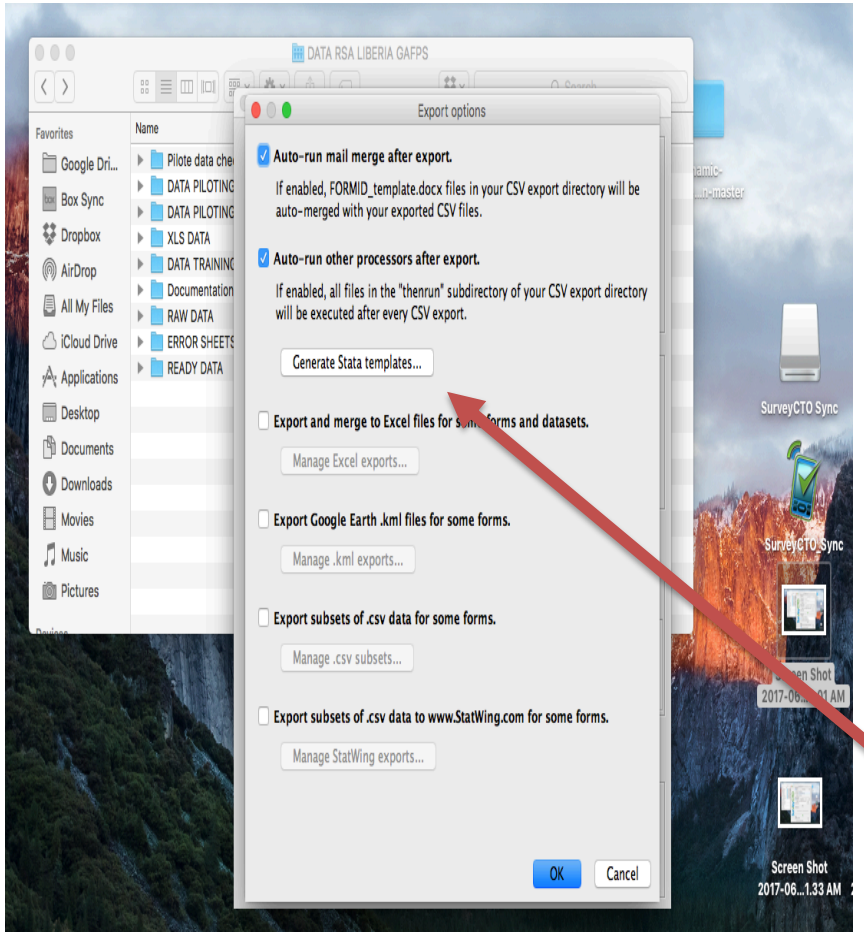

- *SurveyCTO Sync* is the data download tool
- It is an application that sits on your local machine, and needs internet access to be able to download data
- When you open it, this is what it looks like
- **Step 1**: Enter in the log-in details and the folder you would like to download the file to. Then click GO!

# 1. Import SurveyCTO data to Stata



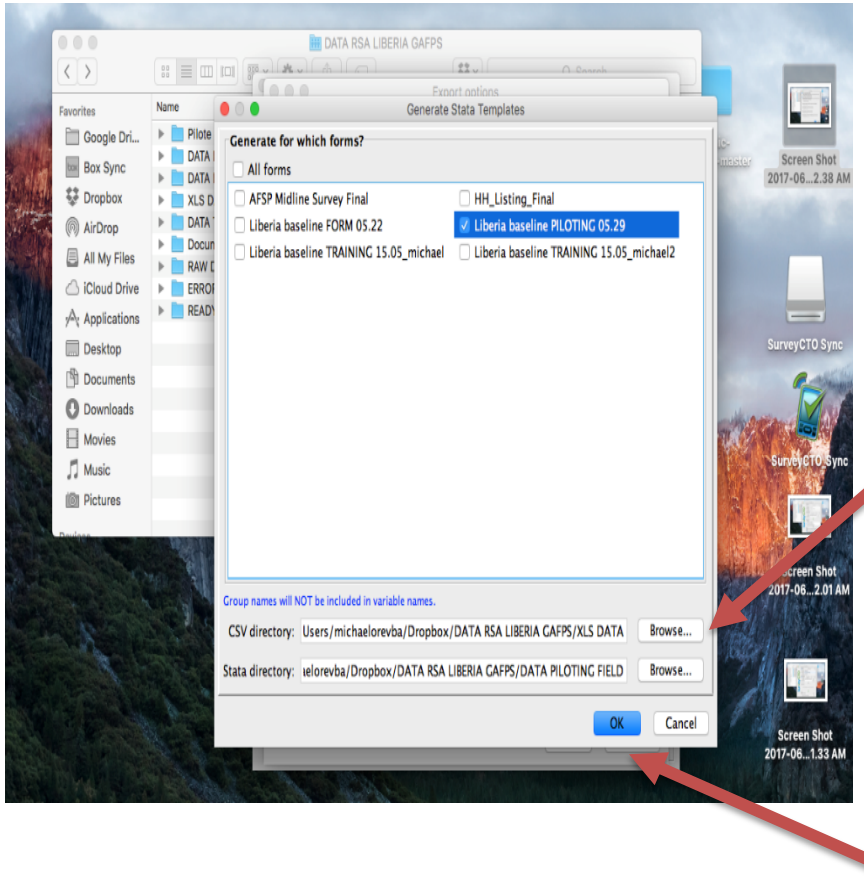

- All of the forms that contain data will show up in the window at the bottom
- You may need to download all of them, or a single one
- In this example, I'm choosing to download just the pilot data.
- **Step 2:** Select the form you want to download AND select the "Show export options after download" option at the bottom
  - Note: you can always select this option later using the taskbar, but this will save you time

# 1. Import SurveyCTO data to Stata



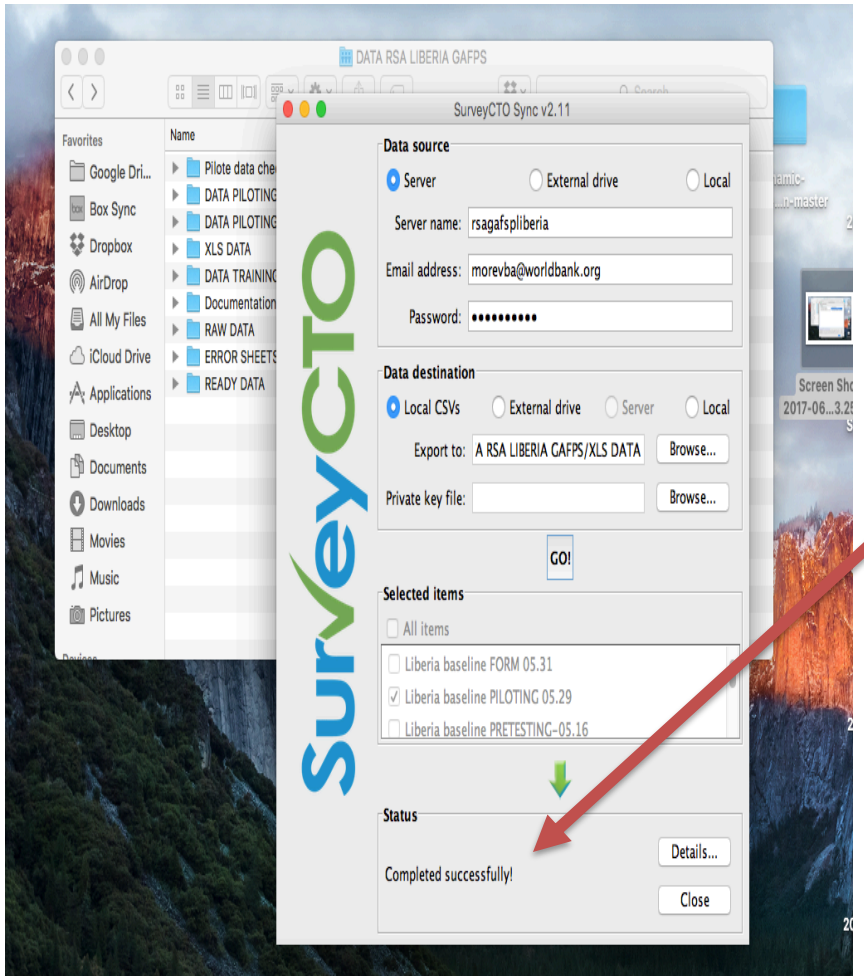

- The ‘export options’ contain a number of different possibilities
- The one I wanted to show you about today is the STATA template that CTO generates to help set up your dataset
- **Step 3:** Click on the ‘Generate Stata templates…’ option

# 1. Import SurveyCTO data to Stata



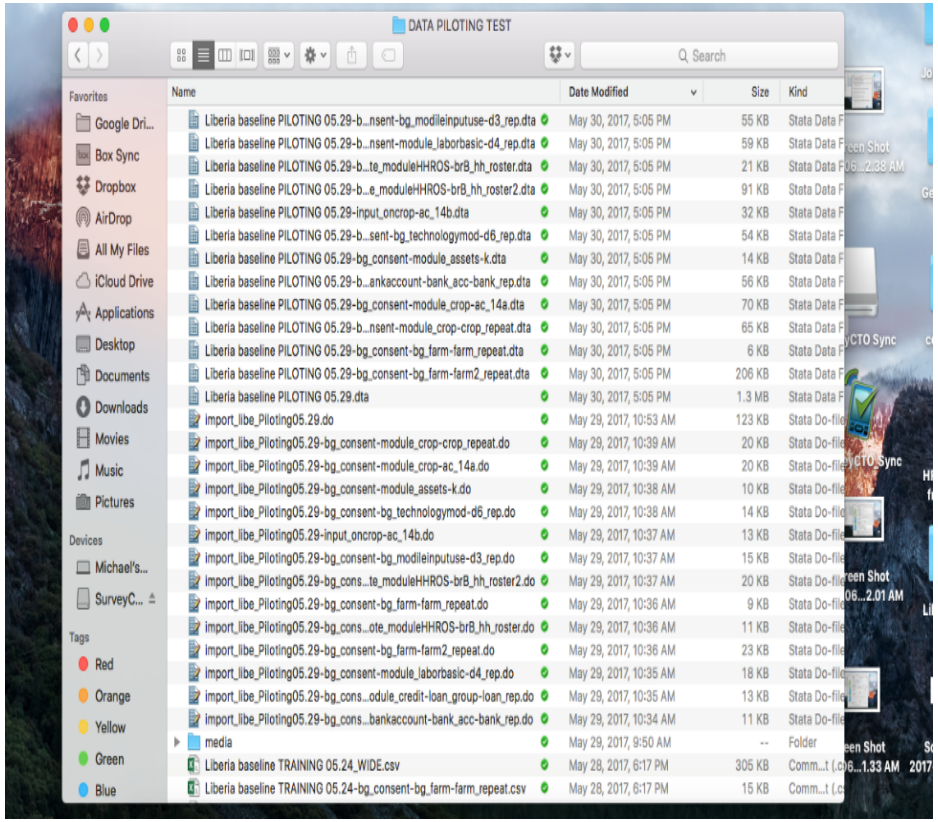

- Once again, a number of different forms come up. Note that this includes forms not only from your current server, but your past work as well (this example has an AFSP project, as well as the Liberia Project)
- Another important aspect of this is what you choose as your CSV directory (the raw-est data) and your STATA directory
  - This helps CTO write the do-file, so as to know where to open the raw data from, and where to save its DTA to
- Remember that we had already chosen the CSV directory earlier in the process, so this step is really about choosing the DTA location.
- **Step 4:** After you have selected the folder, click ‘OK’

# 1. Import SurveyCTO data to Stata



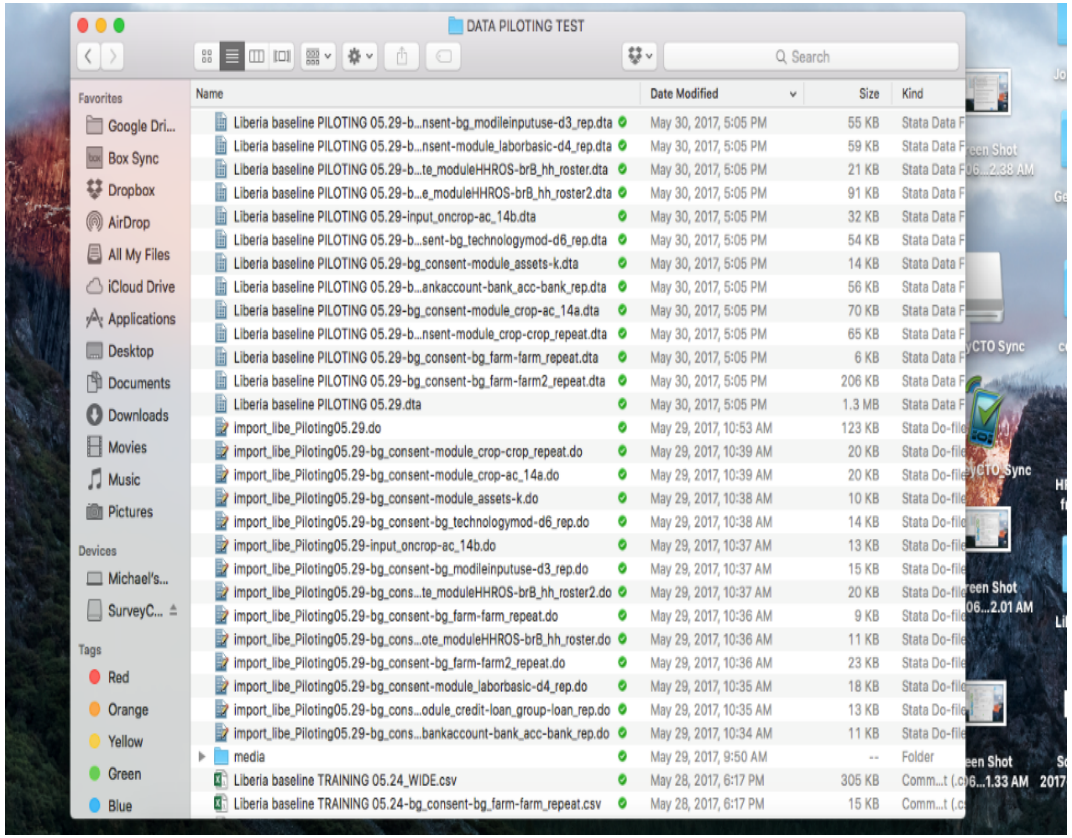

If everything has gone according to plan, you will see a ‘Completed successfully!’ message

# 1. Importing SurveyCTO data to Stata




- CTO will now export do-files into whatever folder you have asked it to export the dta-files into
  - Remember that in this case CSVs are also being exported into that same folder
- Each repeat group has it's own do-file, but there is also one do-file for the 'master' dataset i.e. all of the questions throughout your form that are not within any of the repeat groups
  - Nested repeat groups will each have their own do-files too
- **Step 5:** Open your do-files and run them!

# 1. Importing SurveyCTO data to Stata




- .dta-files will now (not so) magically appear in that same destination folder!
- Voila!

# Import Data to Stata

- The data will be outputted in .csv format.
- Always use SurveyCTO's Stata template import do-file (see earlier slides for instructions on how to download it)
- This will save you days of work!

# 2. Duplicates

- **Purpose:** NONE! They're there to make your life miserable

- **Best to solve in real time!**
  - Much easer to solve the day after interview
    - Enumerator still remembers the interview
    - Field team still close to the respondent and can go back
  - We want to make sure all interviews are uploaded to the server. The only way to know how many interviews we have on the server is to first look for duplicates
  - Other quality checks (back checks, high frequency checks, etc.) depend on uniquely identifying ID variables or are biased by duplicates

# ietoolkit

- A package of commands that we have developed with impact evaluation needs in mind
  - Can be used in many non-impact evaluation projects as well
- Install in Stata by typing: `ssc install ietoolkit`

- For more details see: https://github.com/worldbank/ietoolkit
- We have recently started to work on `iefieldkit` with commands developed specifically for data work in the field

# ieduplicates

- Developed especially for, but not limited to, data downloaded from SurveyCTO servers
- Outputs a report in Excel. The report is also used to correct the duplicates in Stata
- Field supervisors without knowledge of Stata can make the corrections in the Excel file. The duplicates will be corrected next time you run the code
- The command always return the data set with the ID variable uniquely and fully identifying all observations

# ieduplicates - output

| | A | B | C | D | E | F | G | H | I | J | K |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | hhid | dupListID | dateListed | dateFixed | correct | drop | newID | initials | notes | KEY | |
| 2 | 710605 | 1 | 13Nov2015 | 13Nov2015 | yes | | | kb | | uuid:b74fa596-4a3a-43c | |
| 3 | 710605 | 2 | 13Nov2015 | 13Nov2015 | | yes | | kb | they are a | uuid:05e8e695-a090-496 | |
| 4 | 710605 | 3 | 13Nov2015 | 13Nov2015 | | yes | | kb | they are a | uuid:2ec44556-3eb6-403 | |
| 5 | 704604 | 4 | 16Nov2015 | 4Jan2016 | yes | | | ma | | uuid:eeb7249a-e347-48e | |
| 6 | 704604 | 5 | 16Nov2015 | 4Jan2016 | | yes | | ma | they are a | uuid:939f014d-a88c-4413 | |
| 7 | 704906 | 6 | 16Nov2015 | 16Nov2015 | | yes | | kb | they are a | uuid:b4bf0b16-0109-4b8 | |
| 8 | 704906 | 7 | 16Nov2015 | 16Nov2015 | yes | | | kb | | uuid:3975b7cd-d5ad-446 | |
| 9 | 705319 | 8 | 16Nov2015 | 4Jan2016 | yes | | | ma | | uuid:5d1c7b52-e291-4a8 | |
| 10 | 705319 | 9 | 16Nov2015 | 4Jan2016 | | yes | | ma | they are a | uuid:062be7e0-89f8-4aff | |
| 11 | 706223 | 10 | 16Nov2015 | 16Nov2015 | yes | | | kb | | uuid:dd1e682b-d860-40b | |
| 12 | 706223 | 11 | 16Nov2015 | 16Nov2015 | | yes | | kb | they are a | uuid:81a231a7-e5c5-49a | |
| 13 | 706620 | 12 | 16Nov2015 | 16Nov2015 | | yes | | kb | they are a | uuid:ce201e01-bf08-4b7 | |
| 14 | 706620 | 13 | 16Nov2015 | 16Nov2015 | yes | | | kb | | uuid:28463a66-158d-4ed | |
| 15 | 707818 | 14 | 18Nov2015 | 24Nov2015 | yes | | | ma | this surve | uuid:e40b8f50-0dd2-405 | |
| 16 | 707818 | 15 | 18Nov2015 | 24Nov2015 | | | 707804 | ma | the enum | uuid:ba3513dc-c2ac-4ca | |
| 17 | 708122 | 16 | 18Nov2015 | 18Nov2015 | yes | | | kb | | uuid:4c08d9a0-e13c-4bf | |
| 18 | 708122 | 17 | 18Nov2015 | 18Nov2015 | | yes | | kb | they are a | uuid:b1c1809a-1880-437 | |
| 19 | 701405 | 18 | 24Nov2015 | 24Nov2015 | yes | | | kb | | uuid:e5640ced-8e57-482 | |

# Three main types of duplicates in SurveyCTO

- **Type 1** - Double submissions of same observation and the same data
  - First upload from tablet not complete due to bad internet
  - Enumerator resends the form because he/she didn’t see the « successfuly sent » message
- **Type 2** - Double submissions of same observation but with modified data (rare in SurveyCTO)
  - Answers modified after submission, and resubmitted
  - Bad practice, more transparent to correct in a do-file
- **Type 3** - Incorrectly assigned ID. Two respondents are given the same ID
  - Typo in field when entering respondent ID

# Three main reasons for duplicates in SurvyeCTO

- **Type 1** - Double submissions of same observation and the same data
  - Few variables differs between the duplicates and differences are in submission date
- **Type 2** - Double submissions of same observation but modified data
  - Some variables differs between the duplicates and some of them are in observation data
- **Type 3 -** Incorrectly assigned ID. Two respondents are given the same ID
  - Many variables differs between the duplicates and many of the differences are in observation data

# Resolving duplicates

- Why is this classification useful?
  - The vast majority of the duplicates are of Type 1, and those duplicates are possible to solve by just looking at the data. I.e. you can solve them immediately without further investigation
- How to solve Type 2 and Type 3?
  - These duplicates requires qualitative investigation, but since most are Type 1 you will not have to do investigations that often

# iecompdup

- Compares all variables across a pair of duplicates and provide you with a list of the variables where the duplicates has different values.

- From this information we can decide which type of duplicate
  - If it is type 1 we can solve it immediately.
  - If it is type 3 then we are halfway to the solution
  - Type 2 is rare. Re-evaluate if it is possible type 1 or type 3. Otherwise investigate.

# Task 1 – ieduplicates and iecompdup

1. Run the line of code with ieduplicates, and open the report. Was there any duplicates? Which type are they?
2. Run iecompdup once for each of the IDs, what do you find?
   - Add more variables to the iduplicates report, for example: **enumerator_ID supervisor_ID pl_id_09**
3. Do you find any Type 1 duplicates? Which one should you drop?
   - Correct these duplicates where we have prepared code for that
4. Do you find any duplicates that are not Type 1? What type do you think that is? (We have prepared the correction for this case)
5. Save the data file without duplicates, you will need this data set later

# 3. Survey Log

- Make sure to keep records in the field that is updated very day with how many interviews that were completed

- Make sure that the number of observations on the server matches these records.

- Purpose:
  - Provide your team a quick overview of progress on the field
  - Detect enumerators who are slacking
  - Check balance if this is important for the survey (e.g. by gender)

# Task 2 - Survey Log

- Run section 2.1 of the Task 2 code
  - Let's discuss these results together

- Run section 2.2 of the Task 2 code
  - Let's look at the output together

# 4. Other High Frequency Checks

- Checks to run on daily basis on the data collected
- These checks provide information on :
  - the quality of the programming,
  - the quality of the data,
  - Enumerator performance and survey progress,
  - Data flow, distribution and trends across survey
- Share your HFC report with the team for corrections

# Daily checks

**Routine and logic checks**

- All interview should be completed
- Double check key skip patterns
- Double check important hard checks
- Check that no variables have only missing values, where missing indicates a skip

**Date checks:**

- Check start & end date of interview are the same
- Start date and time < end date and time
- Check duration

**Unique ID**

- Some variable must be unique: Respondent_ID, starting date and time

**Consistency checks:**

- Logic checks not implemented in the programming

# Weekly checks

**Distribution Checks**

- % of missing values, « don't know » or « refuse to answer » to each variable
- Check outliers & extreme values
- Review other specify to ensure consistency of answers entered
- Check the range limits programmed in the form have not been reached

**Enumerator Performances**

•ALWAYS review enumerator comments!

•Check number of survey completed by enumerators

•Check average survey duration per enumerators

•Check % of *don't know, zero, no, refuse to answer* per enumerators

•Check enumerator responses on key variables

# HFC – today's checks

- Not enough time to go over all types of checks
- We will focus on discussing and interpreting the results, rather than writing the checks
- There are templates that help you write these checks and help you remember which checks
  - IPA - https://github.com/PovertyAction/high-frequency-checks
  - DIME is working on our own template
- See track 2 for more details

# 5. Back checks

- **Purpose**
  - To monitor the quality of field work: Do some enumerators need extra support from supervisors?
  - To understand whether your questionnaire accurately captures the key outcomes of your study: Do respondents understand the questions the way we intended to?

- **Best practices:**
  - ~ 10% of surveys, 20% in the first 2 weeks of field work
  - Every team and every surveyor must be back checked
  - The back check sample must include a proportional number of missing and replacement respondents.
  - Selection of households for back checks must be random

# Selecting Back Check Questions

- **Type 1 variables**
  - Straightforward questions where we expect very little variation
    - E.g. education level, marital status, occupation, has children
- **Type 2 variables**
  - Questions where we expect capable enumerators to get the true answer
- **Type 3 variables**
  - Questions that we expect to be difficult.
    - Want to understand if these questions were interpreted in the field as intended
- **Identifying Respondent and Interview Information**
  - Check if we have the right person; if the interview took place & when
- **Total duration**: 10-15 min

# bcstats

- *ssc install bcstats, replace*
- **Purpose:**
  - compares back check data and survey data, producing a data set of comparisons
  - completes enumerator checks and stability checks for variables

# Using bcstats report to improve data quality

- **Type 1**
  - If >10% discrepancies → give surveyor a warning
  - 20-30% discrepancies → 2nd backcheck to determine who made errors
    - If error is by surveyor → audit 3 additional surveys by that surveyor in same week
      - if 20-30% discrepancies → put him/her on probation
  - >40% discrepancies → 2nd backcheck to determine who made errors and maybe re-survey HH
    - If surveyor made errors → re-survey HH & audit all surveys done by surveyor in this batch
      - If one more survey has 40% discrepancies → fire surveyor immediately & redo all his/her surveys with 20% or more discrepancies

# Using bcstats report to improve data quality

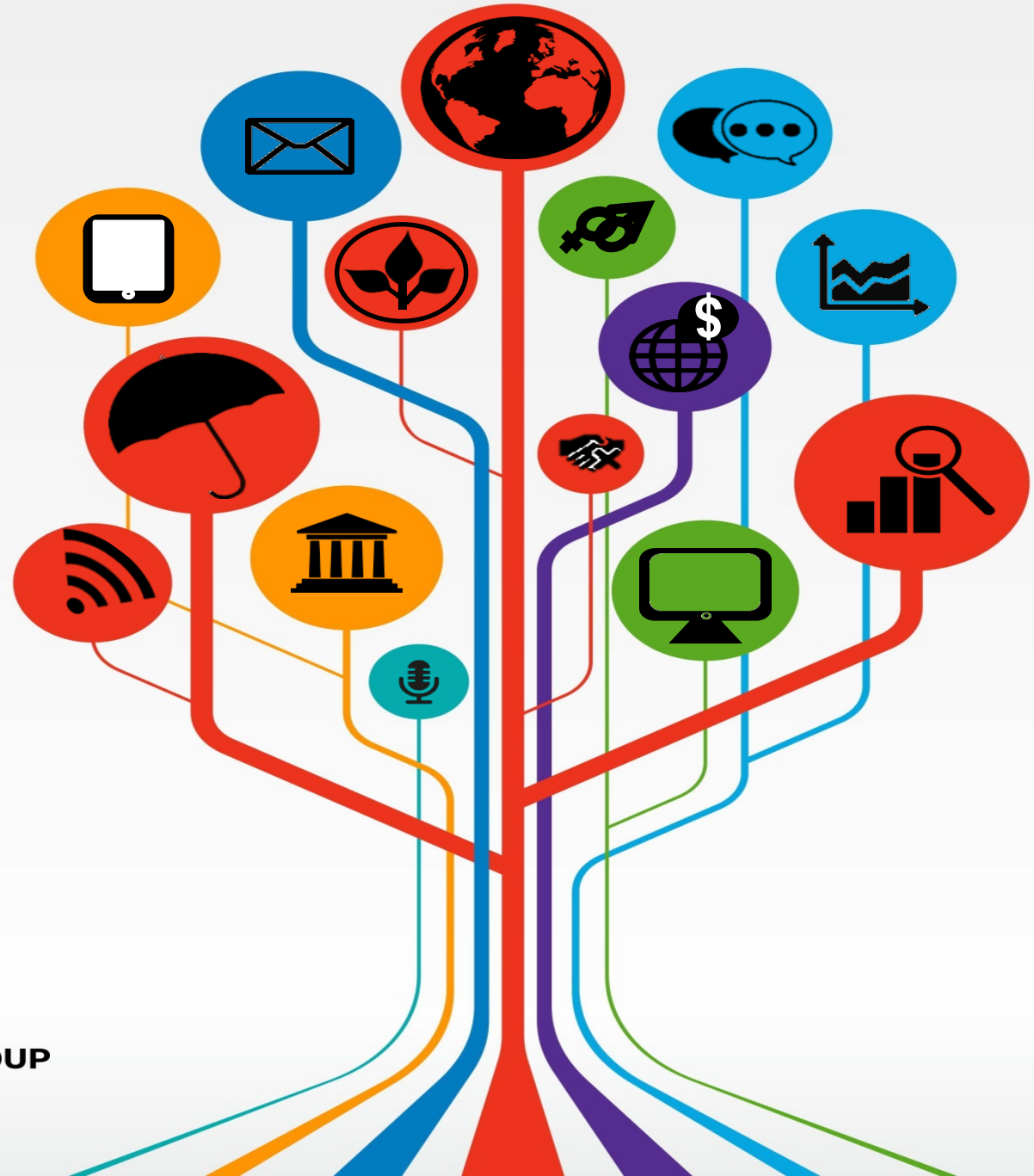
- **Type 2**
  - >10% discrepancies -> Consider re-training
  - If one surveyor is responsible for more than 30% of the errors in a single question, follow the steps for Type 1

- **Type 3**
  - Discuss >10% with your survey team and your PIs.
    - PIs may decide to edit survey or add additional rounds

# Thank you!

Aurelie Rigaud & Kristoffer Bjärkefur

20 June 2018



i2i DIME
TRANSFORM DEVELOPMENT

WORLD BANK GROUP