# User Guide

## Contents

* [Introduction](#introduction)
* [Purpose](#purpose)
* [Interface](#interface)
* [Behaviour Notes](#behaviour-notes)
* [Messages](#messages)

# Introduction

This document describes how to use the ODK Aggregation Tool (OAT), and provides information on the types of errors that can occur and how to resolve them.

# Purpose

The purpose of the OAT is to enable the preparation of ODK Collect data for analysis in Stata.

The OAT reads the instance XML files produced by ODK Collect, and the XLSForms used to prepare the XForms, and produces an Stata XML file. The Stata XML file includes all the instance XML data, with the metadata such as variable labels, types, formats and value labels. One Stata XML file is produced per XLSForm definition (including all versions).

Since the OAT is a thin wrapper around modules that do the work, the underlying code could either be included in other libraries or extended to support other export formats. Similarly, Stata has a command line control facility for automating analysis or conversion procedures.

# Interface

The user interface presents 3 text boxes for selecting the following inputs:

* "XLSForm definitions path": the folder containing the XLSForm XLSX files to read. This folder should contain at least one XLSForm, but ideally contains a copy of all versions of the XLSForm that were used to collect data.
* "XForm data path": the folder containing the XForm instance XML files to read. This folder should contain at least one XML file.
* "Output path": the folder to save the Stata XML file(s). Additionally, if a large amount of log messages are produced during processing, a "log.txt" file will be saved in this location with a copy of the messages.

The above paths can be either:

* Selected by clicking the corresponding "Browse..." button to the left of the text box
* Copying and pasting in the path
* Typing in the path

Once these inputs have been entered, click the "Run" button to execute the aggregation task. When the task is initiated, a message will be written to the "Last run output" textbox to confirm that the task has started. Any further messages, for example informational or error messages, will be written to the same output textbox. More detail on these messages is included in the below section: "Messages".

Each Stata XML file can be read into Stata using the following command:

xmluse "//path/to/stata/xml/file.xml", doctype(dta)

Stata can then be used to analyse the date, or export it to a wide variety of formats.

# Behaviour Notes

If the amount and/or size of XML files being processed is very large, the OAT may become unresponsive while processing is happening in the background. Once a result has been reached, the output messages will be displayed.

Where multiple versions of a XLSForm have been read, the definitions will be sorted (numerically or alphabetically), and the most recent definition for each variable will be used in the output. For example, in version 1, an item might say "What colour do you like?" and version 2 updated the item to say "What colour do you REALLY like?" - in this case, the version 2 definition is used. This also allows metadata for items that appeared in version 1 but not version 2 to still be included in the output.

If the XForm files include duplicates, the first discovered copy (highest in directory tree / alphabetically) will be used in the output. A message will be shown to indicate which files are duplicates and which one was used. The determination of a "duplicate" is based on the XML file content being exactly the same between one or more files.

Additional notes about output content:

* XML elements read from instance XML files that don't have any matching XLSForm definition will be included as a text variable.
* XML attributes other than the form\_id "@id" and form version "@version" will not be included in the output.
* Currently, only the following mappings for XLSForm variable types to Stata data types and formats are included: "start" (str26, %26s), "end" (str26, %26s), "deviceid" (str17, %17s), "date" (int, %td), "text" (str2045, %30s), and "integer" (int, %10.0g)
* Date variables are converted to the Stata Internal Format (SIF) which is an integer representing the number of days (positive or negative) between the specified date, and the Stata zero date of "1960-01-01".
* Metadata is read from the following XLSForm locations for Stata purposes, each is assumed to have content that is valid for each use in Stata (e.g. contains valid characters):
  + "name": used for the Stata variable name
  + "type": used to determine Stata data type and format, per the above mappings.
  + "label": used for variable labels. If a (non-standard) column named "name\_description" is included, this is used in preference to the label column.
  + "choices" sheet: used for value label definitions

# Messages

The following types of messages may be shown, and are listed in approximately the order that they might be expected to be encountered.

Error messages are indicated by "ERROR" - in general these problems must be resolved. Informational messages are indicated by "INFO" - these indicate the performance of normal behaviour and may be used to help understand the nature of the output.

* ERROR: "*input* is empty ...": the named input textbox requires a value but it is currently empty.
* ERROR: "*input* does not correspond to an existing directory": the named input textbox has a value which doesn't correspond to a directory that exists, or is accessible.
* ERROR: " ... while trying to read the XLSX file at the following path ...": the named XLSX file could not be read. It may be open (e.g. "Permission denied" error), or an invalid file (e.g. "Corrupt file" error). Try to collect the known good, valid XLSForms into a separate directory and use that instead.
* ERROR: "The required sheets for an XLSForm definition ... were not found": the input XLSForm directoy included XLSX files that did not contain the sheets required for a valid XLSForm definition - as above, try to collect the known good, valid XLSForms into a separate directory and use that instead.
* ERROR: "No XLSForms were read from the specified path": try to collect the known good, valid XLSForms into a separate directory and use that instead.
* INFO: "The following XLSForm form\_ids were read": these XLSForm definitions will be processed.
* INFO: "Reading XLSForms for form\_id *id*, sorted by version in order of *versions*": the metadata is being preferentially read from the XLSForm versions in the stated order. If there are missing versions, try to collect the known good, valid XLSForms into a separate directory and use that instead.
* INFO: "Removed XML attributes from parsed data ...": the only XML attributes included in the output are "@id" (form\_id) and "@version" (form version). Any attributes listed in this message were removed.
* INFO: "Found duplicate XML files ...": the listed XML files appeared to be exact duplicates, and only the first listed will be included in the output. Check the named XML files and remove any genuine duplicates, or adjust the "XForm data path" input parameter to be more specific to a location without duplicates.
* INFO: "Data was read for the following variables, but there was no metadata for them in the XLSForms ...": the listed variables were found in the XML data but there wasn't any definitions available in the processed XLSForms. This will mean that output file will include the variable values as text, without any particular labelling. If there is a XLSForm definition specifying metadata for this variable, include it in the input path selected for "XLSForm definitions path".
* INFO: "Metadata added to form\_id *id* for the following unknown variables ...": the listed variables were successfully added to the form definition for the purposes of further processing; all this means is that they will be included in the output for the current task run.
* INFO: "Removed variable metadata for the following fields assumed to be empty labelling variables (type=text and read\_only=yes)": since the XLSForm and XML data includes elements for variables that are text and don't accept input, but are included for labelling purposes. These variables are removed from the output data as they contain no information.
* INFO: "Using default language: *lang*, for metadata with form\_id *id*": in the XLSForms settings sheet, a default language is specified. If multiple languages are defined for the variable or value labels, this default language is used for the Stata XML output.
* INFO: "Found non-integer value in choice list: *name*, value: *value*. Variables using this choice list will not have labelling applied as Stata only allows labelling integer values.": the stated choice list included a value that could not be converted to an integer. Stata only allows value labelling of integer values, so values for variables that use that choice list will not have labelling applied..
* INFO: "Collected data for *count* observations for form\_id: *id*": the stated number of observations were included in the output for the stated form\_id. If this is less than expected, check that the input path for "XForm data path" is correct, and contains all relevant XML files.
* INFO: "Wrote form data for form\_id: *id*, to a file at: *path*": the Stata XML file for the stated form id was written to the stated path.