



# 9 Build a Data Extract or Reporting plug-in

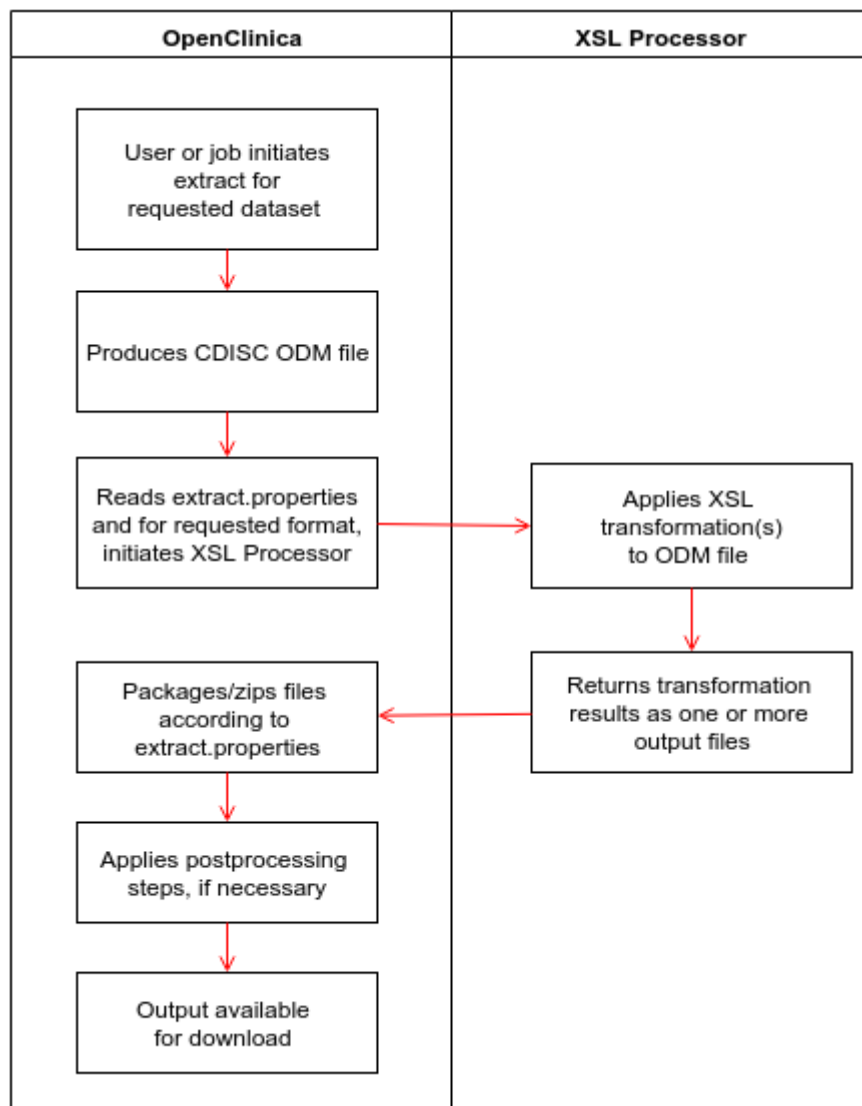
## Overview

OpenClinica's Extract Data architecture lets you develop data extract formats that meet your precise requirements. It does this by:

- Specifying available formats, their associated stylesheets, and associated properties (like filename, archival settings, and whether to compress the file) in a properties file (the [extract.properties](#) file)
- Using XSL stylesheet transformations to read native [CDISC ODM XML](#) and output the data in a transformed format.
- Optionally, enabling postprocessing of the transformed data to output to certain non-text file formats and destinations.

## Add an extract format to your OpenClinica environment





- Locate XSL files for the format you want to add. You can find packages on the [OpenClinica Extensions](#) site and in Lindsay Steven's [Github repository](#).
- Add your files to the xslt directory in your OC environment, normally your\_OC\_data\_directory/xslt
- Edit your [extract.properties](#) and add a new extract form.
- Restart OpenClinica and test it out!

## Create a new extract format

You can add your own transformations to get data into a wide variety of formats using the XSLT language.

- Familiarize yourself with OpenClinica's implementation of [CDISC ODM](#).
- Create your XSL file. While you can start from scratch, you'll save time if you work off one of the [existing](#) OpenClinica extract files, from the Extensions site, github, or CDISC's [Define.xsl](#).
- If your requirements include outputting several files at once (such as a data file and load script), look at the SPSS format in extract.properties to see how you can include multiple XSL files and have them produce multiple output files.
- Postprocessing: To do things that XSLT cannot do by itself, like produce PDF files or load the data into external relational databases for ad-hoc reporting, a postprocessor framework is available to generate binary output formats or send data to a target destination. Two postprocessors are included: output to a database using JDBC connectivity and generate PDF

files using XSL-FO. The postprocessing step is transparent to end-users; they simply get their files for download or alternatively receive a message that the data has been loaded into the database. Instructions for use are provided in the `extract.properties`.

- Add the XSL to your OpenClinica environment as described above.

## Use your extract format

Initiate an extract for your study from the [Download Data](#) screen or via a [job](#) and select your new output format. Execution follows a five step process:

1. OpenClinica generates CDISC ODM XML version 1.3 with OpenClinica Extensions
2. OpenClinica applies the XSL transformation and generates output file(s) according to the settings in `extract.properties` for the specified format
3. Optionally, if postprocessing is enabled for the requested format, OpenClinica runs the post processing action according to the settings in `extract.properties`.
4. OpenClinica provides user notification with success or failure message.
5. The data is available for download.

## Other notes

- A framework exists in the code to add additional postprocessors via the addition of Java classes with references to those class names in the `extract.properties` file.
- Do not replace the extract XSLs that come with OpenClinica. If you do, your changes will be overwritten with the original contents every time OpenClinica is restarted.
- The Java code in the OpenClinica Extract Data module outputs study metadata and clinical data in only one format: CDISC ODM (version 1.3, with OpenClinica Extensions). OpenClinica's vendor extensions in the ODM file ensure that we can extract all possible data related to a study and its clinical data, even if not supported by the core ODM standard. This includes audit trail, discrepancy, and electronic signature information.
- Transformations are powered by the Saxon XSLT and XQuery processor.

## Sharing

If you improve an existing extract format, create your own, or add a new postprocessor, please [share it with the community!](#)

Approved for publication by Cal Collins. Signed on 2016-03-11 3:34PM

Not valid unless obtained from the OpenClinica document management system on the day of use.