How to Batch Ingest Files

Before you can batch ingest objects, you will need to have downloaded and installed the Islandora Batch Importer module.

For larger collections, Islandora is able to pull multiple files out of a zipped archive and ingest them into Fedora as a batch. There are a few ways that this can be done. You can upload .zip archives full of:

- content files to be ingested
- XML metadata that can later have content files added to it
- both content files to be ingested and XML metadata to be appended to files
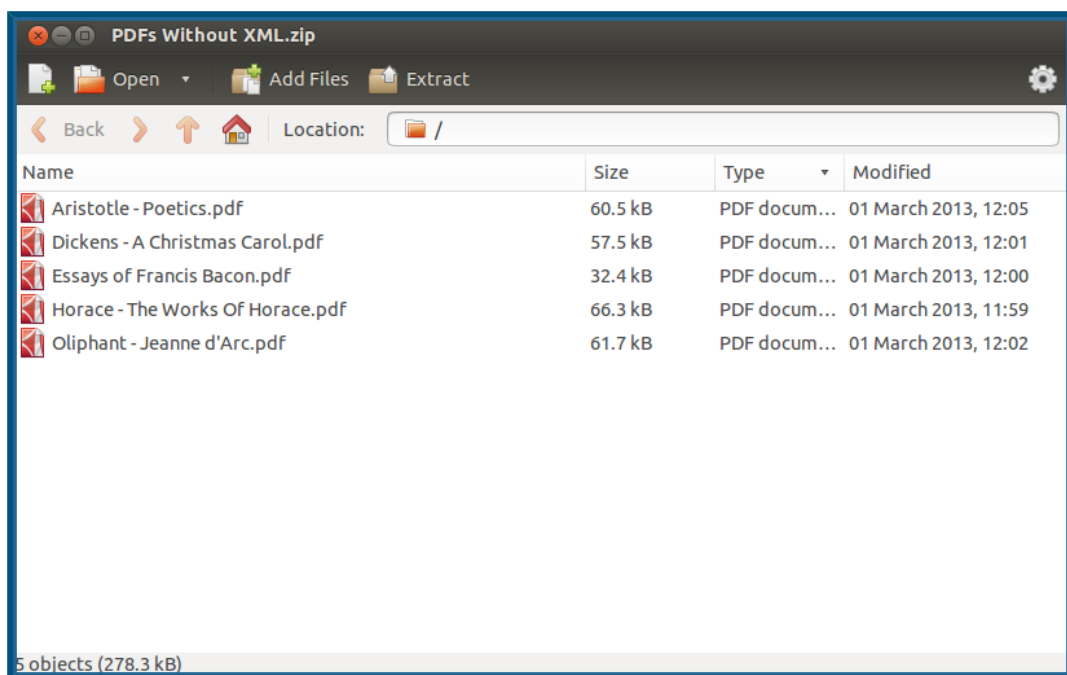- books, formatted with a specific directory structure

This page will run through the specifics of each one. In these examples, we will be batch-ingesting PDF files into a collection with the 'PDF Solution Pack' content model applied to its collection policy.

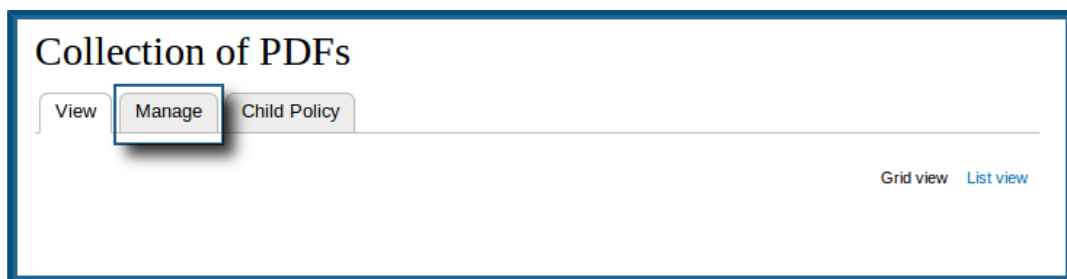Ingesting files from a .zip archive without metadata

1. Create a .zip archive with your files in it

The process for doing this will vary from operating system to operating system, but on PC, Mac and Linux at least, a .zip archive can be made in your file browser by highlighting the file or files you would like to zip, right-clicking, and finding an option similar to 'compress', 'create archive', 'create zipped folder', and so on.

In our example, opening the zipped archive shows our PDFs grouped together:
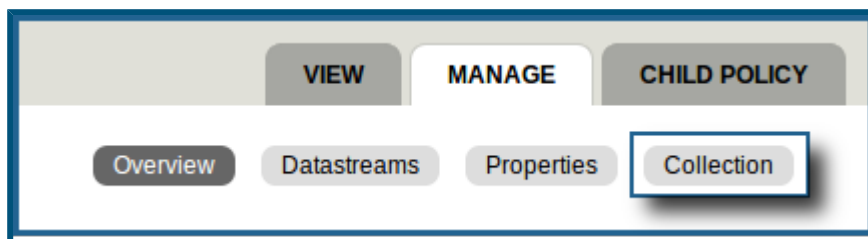
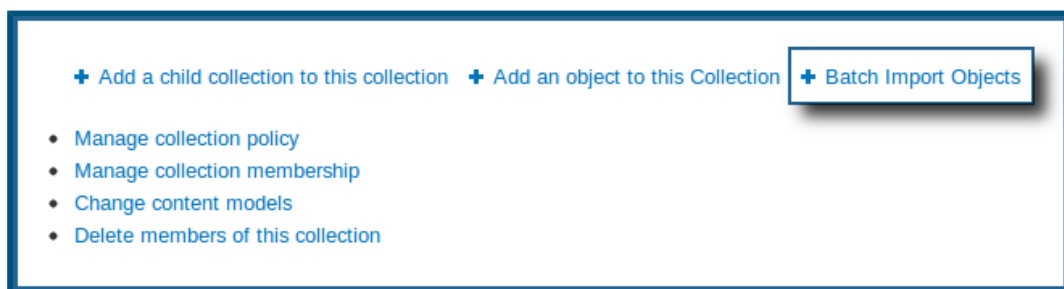**2. Navigate to the destination collection and click 'Manage'**

This will take you to the collection's management page.

3. Click on the 'Collection' button

This will take you to that collection's options page.

4. Click on 'Batch Import Objects'

This will start the batch import process.

5. If other batch import modules are active, choose 'ZIP File Importer' from the drop-down menu

The Islandora Batch Importer module comes with several different modules for handling different types of content. If more than one is enabled, you will need to select the correct one.

In this case, we will be importing objects from a .zip file, so we are going to select that option.

6. Choose the correct options for your batch import

There are a few options on this screen that will need to be set up:

1. Browse to the .zip file you would like to upload, and then click the 'Upload' button. It may take a while to move the file to the server.
2. Choose the content models you would like to apply to the objects.
3. Choose the namespace to be applied to the objects. ("Islandora" is given only as an example.)
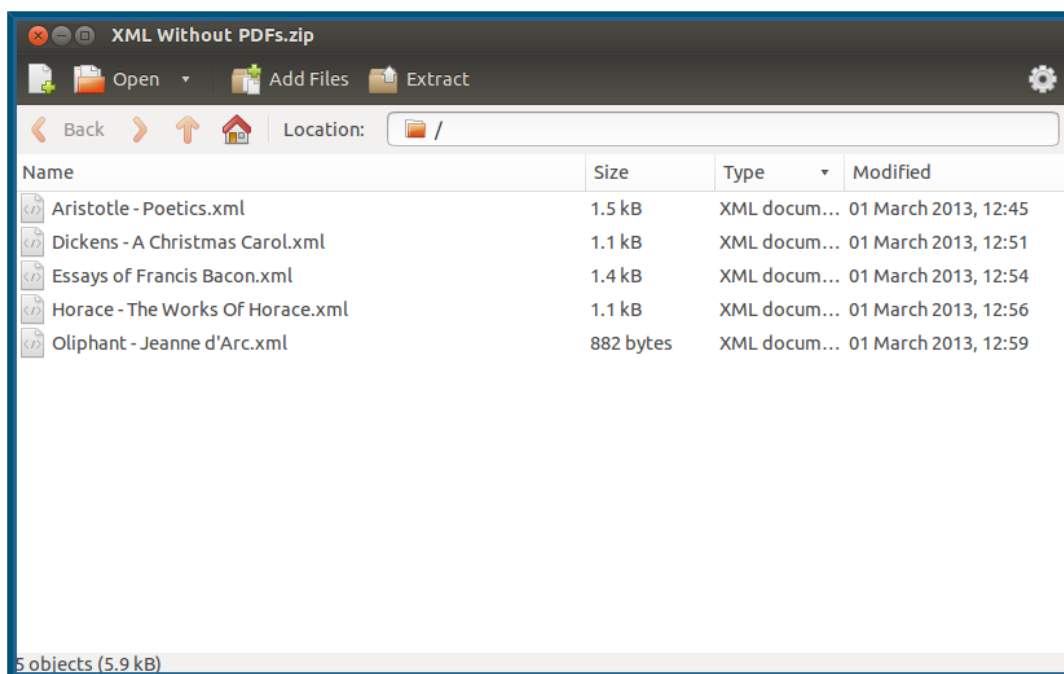
4. Click the 'Import' button to begin the batch import process.

This will import all the files from your zipped archive during which new Fedora objects are created and associated with the specified collection.

Ingesting metadata from a .zip archive

If you wish to ingest objects as simply metadata without a file datastream attached, you may do so by using the batch importer to import a .zip file full of XML forms.

In our example, a .zip file full of XML files has been created. Once the metadata files are fully ingested, PDF files can be added to the objects as datastreams.

In this case, we can simply follow the same steps as in the previous example to perform the batch ingest.

Most Solution Packs look for a file with a MIME type associated with the corresponding content model. In these cases, when uploading XML-only .zip files, the Book Batch Importer will show an error saying that derivatives could not be created. This can be ignored in these cases.

To create the XML files, you can either design them manually in a text editor or XML editor or use the Form Builder built into Islandora.

To use the Form Builder to create XML records, navigate to http://**path.to.your.site**/admin /islandora/xmlform, find the type of form you would like to fill out, and click the 'view' link beside it:

TITLE	TYPE	OPERATIONS			
Audio MODS form	Built-in	Copy	View	Export	Associate
Basic image MODS form	Built-in	Copy	View	Export	Associate
Citation MODS form	Built-in	Copy	View	Export	Associate
Islandora Book MODS Form	Built-in	Copy	View	Export	Associate
Large image MODS form	Built-in	Copy	View	Export	Associate
Newspaper MODS form	Built-in	Copy	View	Export	Associate
PDF MODS form	Built-in	Copy	View	Export	Associate
Video MODS form	Built-in	Copy	View	Export	Associate

In this case, we will be creating an XML file for a PDF.

Fill out the form with the metadata values you would like, and click the 'Submit' button at the bottom of the form. This will output raw XML to your browser that you can then paste into a text editor, similar to the following:

```xml
<?xml version="1.0"?>
<mods xmlns="http://www.loc.gov/mods/v3" xmlns:mods="http://www.loc.gov/mods/v3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xlink="http://www.w3.org/1999/xlink">
  <titleInfo>
    <title>Poetics</title>
    <subTitle/>
  </titleInfo>
  <name type="personal">
    <namePart>Aristotle</namePart>
    <role>
      <roleTerm authority="marorelator" type="text">Author</roleTerm>
    </role>
  </name>
  <typeOfResource>text</typeOfResource>
  <abstract>Aristotle's philosophical treatise on the subjects and methods of poetry.</abstract>
  <genre>article</genre>
  <note/>
  <subject>
    <topic/>
    <geographic/>
    <temporal/>
  </subject>
  <location>
    <url/>
  </location>
</mods>
```
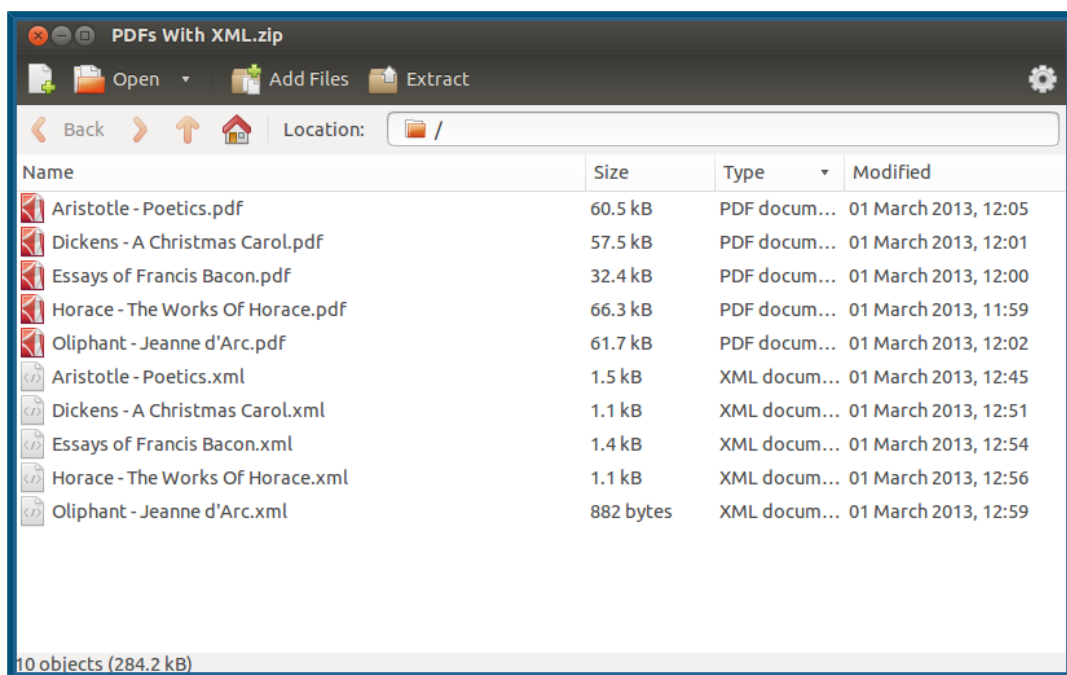
Save this output as an XML file, and add it to your .zip archive.

Make sure you choose the correct content model that you would like to be associated with the files that will eventually be ingested

Ingesting files and corresponding metadata from a .zip archive

Zip archives can contain both the content files to be ingested and the metadata files at the same time. This can be accomplished in the same way as the first example, with a few changes to the .zip archive itself:

In the above example, you will notice that each PDF file has a corresponding XML file, and that the filenames of the PDFs and the XML files are identical in every way – including capitalization – except for the extension. Files with matching filenames will be ingested together into the same object.

After creating a .zip archive like above, you can simply follow the steps from the first example to ingest the batch into the repository.

Not every file in the archive needs to have corresponding datastreams. You could potentially upload an archive that contains some objects without metadata, some objects with only metadata, and some objects with both.