

Batch-uploading materials

For this process, you will need:



- An admin account for [Scholars Junction](#)
- A collection for your items in Scholars Junction
 - Items can only live in *collections*; can't be uploaded into *communities*
- The [IR metadata template](#)
- Some basic knowledge of how to use command line
- [SAFBuilder](#) (This is rather tricky to install, so you will need to ask Systems to install it for you)

Batch-uploading material into DSpace requires creating a metadata file and a special folder structure ([Simple Archive Format](#)), zipping the results, and uploading the zipped files into DSpace.

Step 1: Select or create a collection

1. If you are uploading files into an existing collection, locate it.
2. If you are uploading a new collection, you will need to create a space for it in Scholars Junction.

Step 2: Collect files

1. Save all of the files you're uploading in a single folder on your computer.



1. No subdirectories!
2. This needs to be on your local machine - don't do this process using a shared or network drive.
3. For best results, keep the folder as top-level as possible.

Step 3: Create metadata

1. Use the IR metadata template to complete metadata for the collection, following the instructions in the template. Save this file in the same folder as the files you're uploading.
2. When metadata is complete, save the sheet as `.csv`.
3. If the collection is large, split the `.csv` file into batches of about 25-50 items per `.csv`. This is a trial-and-error process; you may have to edit these files several more times before they are small enough or contain few enough changes to be uploaded to DSpace.

Step 4: Package the data

1. Open your command prompt. (You can click in the "Type here to search" area of your taskbar and type simply `cmd` and it should pop up. Once it's open you can pin it to your taskbar for later ease of use.) When Command Prompt is open, you are in your users folder (you should see `C:\Users\[computer username]>`)
2. Navigate to the folder where SAFBuilder is installed. To do this, type `cd` and the names of the folders above it, hitting Enter at the end of each line, drilling down until you hit that folder.
 - a. Example: SAFBuilder is installed in the Documents folder

```
cd Documents
cd SAFBuilder
```
3. Build the SimpleArchiveFormat package with SAFBuilder, like so:

```
safbuilder.sh -c (filepath) -o (folder name) -z
```

Common Command Prompt commands



Command	Explanation
<code>cd</code>	Change Directory. This moves you into a different location on your computer.
<code>cd ..</code>	Go up a level.
<code>dir</code>	Shows directory contents. Handy if you don't know where to move next and want to see what's in a folder.

Command	Explanation
<code>safbuilder.sh</code>	Calls the SAFBuilder program.
<code>-c</code>	Calls in the metadata <code>.csv</code> file.
<code>(filepath)</code>	Path to the metadata <code>.csv</code> file, including its name. (Example: <code>C:\Users\jdunn\Documents\collection\metadata.csv</code>)
<code>-o</code>	Optional; gives a custom folder name to the output directory. (By default, it's named SimpleArchiveFormat.)
<code>(folder name)</code>	A custom folder name. Optional unless you used <code>-o</code> .

<code>-z</code>	Zips the folder.
-----------------	------------------

Example: `safbuilder.sh -c C:\Users\jdunn\Documents\collection\metadata.csv -o SampleCollection -z`

4. Open the folder and check that everything outputted properly. Along with your materials to be uploaded and your original metadata .csv file(s), you should see a new folder named either SimpleArchiveFormat or the custom name you gave it, and a .zip file with the same name.

Step 5: Upload the package

1. Log into Scholars Junction.
2. On the sidebar, click Batch Import (.zip).
3. Choose the collection you're uploading into from the dropdown.
4. Browse for the .zip file you created.
5. Click Upload SimpleArchiveFormat ZIP.

Step 6: Check the collection

Give it a few minutes to fully process, and then go to the collection in ScholarsJunction and make sure everything's there.