

Still in development

# Imaging Workflow

---

1. Turn on camera
2. Wait for EOS to open
3. Check EOS settings
  1. Confirm that **renameimage.exe** is 'registered' to EOS
    1. Preferences > Linked Software > Register > Browse - select **renameimage.exe**
    2. Full path -  
**C:\Users\Image\Documents\PNWHerbaria\Scripts\renameimage\renameimage.exe**
4. Select portal folder
  1. In EOS - Preferences > Destination Folder
  2. Select one of the portal folders **C:\Users\Image\Desktop\Imaging\**
  3. Personal projects and other special collections that are not going on Symbiota should be imaged in the **Random** folder if you want the images backed up to the server (Sassafrass). Any folders in **Random** will be moved as is to the local backup drive, not sorted into portal folders.
5. Take photo
  1. A small box should pop up with the current file name and an place to enter the new name
6. Scan barcode
  1. Barcode should appear as new file name
  2. Click the "rename" button to rename the file
  3. The small window will close after the file is renamed.

Repeat 5. & 6. for all photos

7. Batch edit photos in DPP
8. Convert from CR2 to JPG
  1. Now you are done, make sure to record on paper what specimens you imaged
9. Upload CSV
  1. CSV files ready for upload are on in the CSVLogs/ folder. On workstation 2 at LSU.
  2. After uploading CSV files please move them to the CSVLogged/ folder.
  3. If file names are edited and re-uploaded a new CSV file will be made for the date the original file was uploaded. This CSV can be combined with or replace the older CSV in the CSVLogged folder.

# Behind the scenes

---

Task Scheduler set to run following scripts.

RunOrganize.sh - runs organizeIncomingImages.py at 8PM every day.

organizeIncomingImages.py - checks barcodes in incoming folders, moves to appropriate folder(based on barcode) on storage drive. Put files with names that are too long or too short in BadBarcode folder.

WakeUp.sh - wakes up the computer to sit and wait for 5 minutes at 10PM, this allows the server to connect to the storage drive, in order to copy over files.

rsyncDaily.sh - runs on server to copy files to server and create derivatives. Then it calls LogtoCSV.py to turn logs into csv files to upload to portals.

LogtoCSV.py - takes all rsync logs that were modified in the last 24 hours and creates or adds to csv files that have not been uploaded to a portal yet.