

Russell Library Oral History and Media Unit  
AIP creation script, explanation  
Updated October 22, 2018

The Russell OHM bash script for generating AIPs is called: **aip-creation.sh**

The script executes commands from other programs/scripts, which should already be installed and configured before running the script on the proto-aip source directory. See the list of dependencies in the readme.md files for instructions on which programs to install and links from which to install securely.

If folders were previously bagged, first validate the bags to ensure that fixity has remained unchanged. Then, to turn these bags in aips, the script essentially “breaks” open the bag, transforms its structure and contents to match the definition of what a finished AIP should look like, and then bags it again, validates the bag, and tar/zips the bag.

**The script performs the following operations in order to so:**

1. (If bagged) Validates bags and moves invalid bags to bags-invalid directory
2. (If bagged) Deletes .txt files from aip directories
3. Recursively converts all uppercase characters into lowercase in all directories/subdirectories
4. If a bag, renames 'data' folder to 'objects'
5. Deletes all but pertinent preservation files from each 'objects' folder
6. Makes a metadata subfolder in each aip directory
7. Generates mediainfo.xml of contents in 'objects' folder of each aip directory and saves it in each 'metadata' folder
8. Makes copy of each mediainfo.xml in the mediainfo-xml folder in source directory
9. Runs mediainfo-cleanup.xsl stylesheet on the mediainfo.xml
10. Runs mediainfo-to-master.xsl stylesheet on the output the previous step, and saves output as master.xml in each metadata folder
11. Copies master.xml files into master-xml folder in source directory
12. Deletes all hidden .DS\_store from source directory before bagging
13. Runs bagit.py to bag aip directories using md5 and sha-256 checksums
14. Runs 'prepare\_bag' script (developed by Shawn Kiewel, UGA Libraries Application Analyst) to tar and zip each aip directory
15. Moves tarred/zipped files into aips-to-ingest folder in source directory

**To run the script:**

Put the file path to the script followed by the filepath of the source directory (containing the aip directories) into a Terminal window.

The scripts first test that it found the right source directory and only then changes directories to the source directory. This is especially important, since the script contains recursive deletion commands which are irreversible. If unintentionally deleted, files cannot be restored. This test is therefore meant as a safeguard before running the script.

## Types of errors reported in script

As you run the script, you will get “echo” messages telling you what the script is doing in real time. For example, when it has finished creating the MediaInfo XMLs, it will read “MediaInfo files generated.” If the script reports error messages when it is running, you will need to verify what went wrong and repeat steps as necessary.

Note: tests are built into the script to check for the existence of files before beginning to act on them. For example, the stylesheet transformation will only run if a mediainfo.xml file exists in each metadata folder of the aip directory. These built-in checks are meant to reduce the amount of errors that are reported in the Terminal output.

### Mediainfo.xml creation reports error

“Error reported by XML parser: Premature end of file.”

Premature end of file means that there were not any files in the ‘objects’ folder of the proto-aip for MediaInfo to run on. The script did not properly copy over pertinent preservation files into ‘objects’ folder before deleting ‘data’ folder.

As a result, although a mediainfo.xml file was generated, it is a zero byte file.

```
mediainfo files copied to mediainfo-xml
```

```
Error on line 1 column 1 of rbrl379amp-020_media_mediainfo.xml:
SXXP0003: Error reported by XML parser: Premature end of file.
org.xml.sax.SAXParseException; systemId: file:/Volumes/DataHD2/pr
oto-aip-script-error-on-purpose/rbrl379amp-020/metadata/rbrl379am
p-020_media_mediainfo.xml; lineNumber: 1; columnNumber: 1; Premat
ure end of file.
```

### Master.xml creation reports error

Because a zero byte mediainfo.xml file was generated, the script could not run the stylesheet transformation to generate a corresponding master.xml, and therefore it cannot find the file in the metadata folder of the aip.

As a result, you get the following error:

```
cp: rbrl*/metadata/*_master.xml: No such file or directory
```

### Cannot zip tar file

Error is the result of when there is not enough drive space left for the tar file to be zipped. You should make space on the drive, and then redo the tar/zip of the aip.

```
bzip2: I/O or other error, bailing out. Possible reason follows.
bzip2: No space left on device
Input file = rbrl379amp-001.111942260736.tar, output file = rbrl379amp-001.111942260736.tar.bz2
bzip2: Deleting output file rbrl379amp-001.111942260736.tar.bz2, if it exists.
Cannot zip tar file:
```