**Separate Attributes to TIF Script**

Hello! Please use this document to get a basic understand of the script before running or editing it. Please update this document with changes you make.

*Please note, columns and attributes are the same thing and are referenced interchangeably.***What should I read?**

*I want to get an overview of how the script works.*

Essential Parameters, Main Function Calls, attributeToImage, Metadata

*I am running the script for the first time on my machine.*

Essential Parameters

*I am running the script on a new year of TreeMap.*

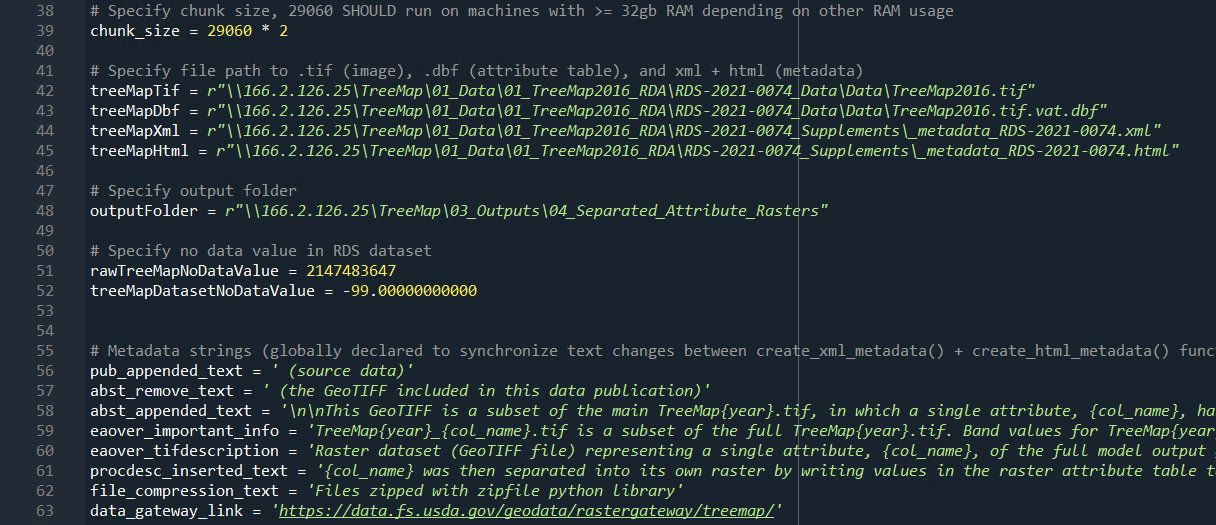
Essential Parameters, Metadata

*I want to run the script on a single attribute / I want to only reprocess the metadata.*

Custom Processing

**Essential Parameters**

Please check these parameters before running the script to ensure proper functioning.



*chunk\_size*

The script processes images in chunks to manage RAM usage. In general, a chunk size of 29060 should be sufficient for systems with 32gb of RAM. 58120 is good for systems with 128gb of RAM. Greater chunk sizes will result in faster processing, however, if available system RAM is exceeded during processing the script will throw an error.

*treeMapTif, treeMapDbf*

The script uses the tif and dbf of the main dataset to separate the attributes. These variables must be filepaths to the main dataset’s tif and dbf. The screenshot shows filepaths to the 2016 version of TreeMap. To separate attributes for different years, these filepaths must be updated (e.g. to the 2020 version of the main dataset).

*treeMapXml, treeMapHtml*

The script uses the xml and html of the main dataset to build metadata for the attributes. These variables must be filepaths to the main dataset’s xml and html. The xml and html are copied and modified for each attribute. These must be updated when processing new years of TreeMap (e.g. TreeMap2020)

*outputFolder*

All outputs (tif, xml, html, arc compatible metadata, arc compatible stats, layer files, readme, zipfiles) will be saved to this folder. Change according to your preferences.

*rawTreeMapNoDataValue*

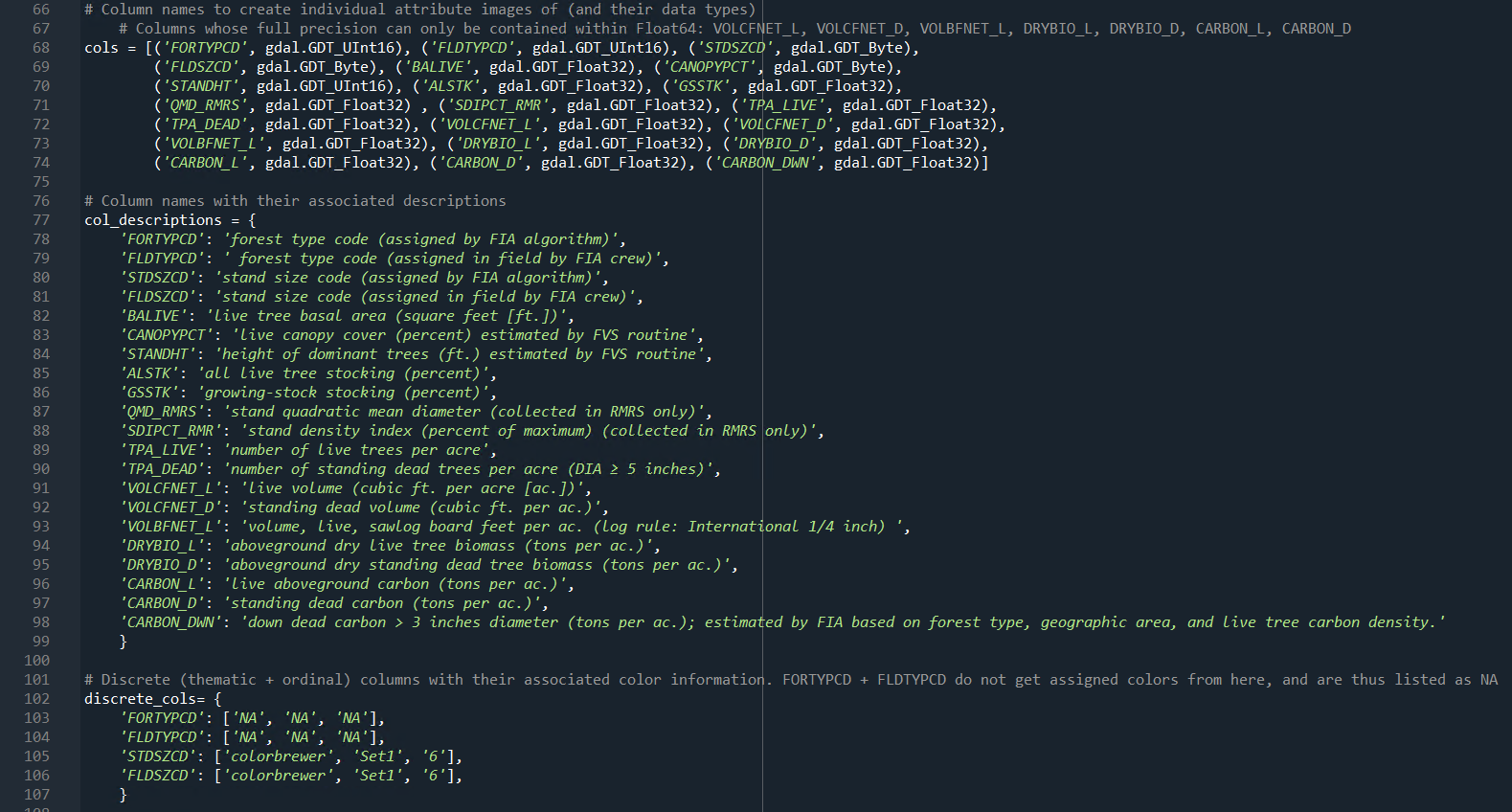
This must be set to the **tif’s raw** no data value. It should not need to be updated unless the raw data value changes between TreeMap years.

*treeMapDatasetNoDataValue*

This must be set to the **dataset’s** no data value. It should not need to be updated unless the dataset’s no data value changes between TreeMap years.

*Metadata strings (pub\_appended\_text – data\_gateway\_link)*

These are strings that get added to or removed from the main dataset’s xml + html to tailor to the new attributes. They are globally declared so they can be managed from one location. They are used in the xml and html metadata functions. Please see the metadata section of this document for more information.



*cols*

This list of tuples is used by the main function to iterate through all the attributes to be separated and assign their pixel values a data type. Update the column names if they change between TreeMap years. Update the gdal data types if you want an attribute’s pixel values to have a different data type. Make sure you understand the implications of losing precision when changing data types.

*col\_descriptions*

This dictionary is used to insert descriptions for each attribute into it’s metadata. Please see the metadata section of this document for more information.

*discrete\_cols*

This dictionary is used to keep track of which attributes are discrete (telling the script which attributes need an attribute table for their symbology). Ordinal attributes (STDSZCD, FLDSZCD) have an associated list identifying the correct color palette in the palettes.json file (gtac-treemap\data\_portal\_scripts\supp\_files\palettes.json). Thematic attributes do not get their color palettes from that file, so they are given ‘NA’ values. Ordinal and thematic attributes are distinguished in the create\_attribute\_table() function.

**Main Function Calls**

These are the default main function calls in the script.

Text

Description automatically generated

*year = determine\_year()*

The year of the dataset is automatically determined by examining the main dataset’s filename (from the variable treeMapTif). This variable gets used throughout the script.

*prompt\_user()*

This function informs the user of the assigned input filepaths and the assigned outputFolder. It then asks them to confirm before proceeding. You may comment this out if you do not want to be prompted.

*for loop*

This is the main for loop. It iterates through each attribute, calls the main function (attributeToImage), and measures the processing time.

*attributeToImage(col\_name, gdal\_dtype)* This function writes a new COG formatted raster for the specified attribute with computed statistics and pyramids. It also calls helper functions to create xml metadata, html metadata, arc compatible metadata, and arc stats files. If the attribute is discrete, it calls a helper function to build an attribute table with an appropriate color palette for symbolization. The function then zips all the associated files together and saves the zip in the output folder.