Project 2: Batch reprojection tool for vector datasets

Some GIS departments have determined a single, standard projection in which to maintain their source data. The raw datasets, however, can be obtained from third parties in other projections. These datasets then need to be reprojected into the department's standard projection. Batch reprojection, or the reprojection of many datasets at once, is a task well suited to scripting.

In this project you'll practice Python fundamentals by writing a script that re-projects the vector datasets in a folder. From this script, you will then create a script tool that can easily be shared with others.

The tool you will write will meet the following requirements:

1. A folder on disk containing vector datasets to be re-projected.
2. The path to a targeted shapefile that has the targeted spatial reference.

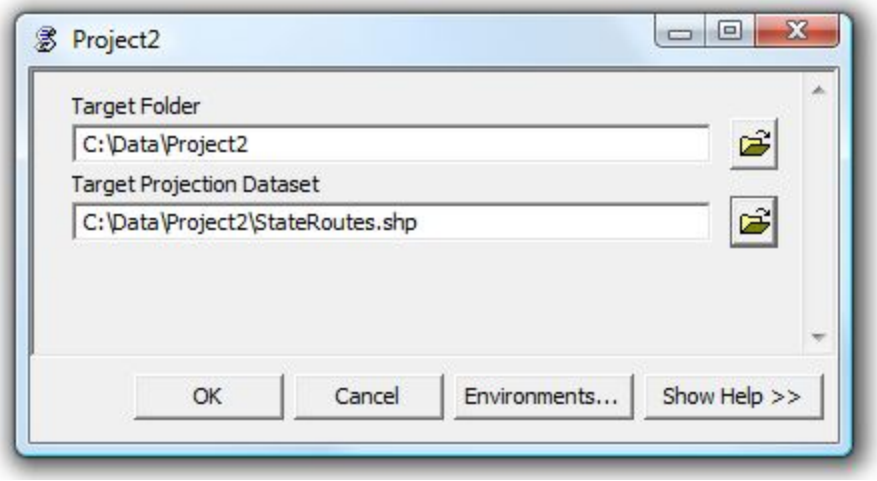

Figure 2.1 The Project 2 tool with two input parameters and no output parameters.

3. If the target dataset is already in the projection, the script will not reproject it, but make a copy of the dataset in the final output folder
4. If the file that has been reprojected has been reprojected, it needs to have "_projected" in the filename.
5. The dataset does not change state. If the targeted dataset is a shapefile, the dataset should remain a shapefile as the output. If the file is a featureclass, it should remain as a featureclass in the output folder
6. A well-documented script that prints messages to the screen with the progress of the script
7. No part of the script will have hard-coded values
8. A short writeup describing what you learned during this project and how you approached the problem.

The beyond the scope (extra credit) portion:

1. Side panel help and documentation
2. Try-except handling
3. Script uses relative paths

Deliverables:

1. A zipped file that contains:
    - Documentation
        - A short (300 words) write-up describing what you learned during this project and how you approached the problem
        - Flowchart
        - Tool Helpfile (optional)
    - ToolData
        - Final Output Folder
            - FinalData.gdb
            - Shapefiles
    - Scripts
        - Your Script (.py file)
    - Project2.tbx
        - <your_name>_BatchProject

Tips and Tricks:

- Remember about string concatenation: This will help you when you are renaming your file.
- Do Not use the Esri Batch Project Tool on this project- you're going to make your own variation of a batch projection tool by running the Project tool inside a loop.
- There are many ways to remove ".shp" from the filename. you could use syntax like this:

```
rootName = ""
if fc.endswith(".shp"):
      rootName = fc[:-4]
```

- To check and see if the dataset is already in the target projection, you need to obtain a spatial reference object for each dataset, and then compare the spatial reference names of the two datasets. (remember: arcpy.Describe().Name). Don't compare the spatial reference objects themselves, because two spatial reference objects could have the same name property, but be different.
- You will want to add messages at each major step of the way. Remember to use Arcpy.AddMessage(arcpy.GetMessages()) immediately after the line where you run the Project tool. Each time the loop runs, it will add the messages from the current run of the Project tool into the results window. If you wait to add this line until the end of your script, you only get the messages from the last run of the tool, so it's important to put the line inside the loop. See https://www.e-education.psu.edu/geog485/node/127 for more detail.
- If you need extra help with making the script tool, refer to the PSU Lesson 1.7.1 at https://www.e-education.psu.edu/geog485/node/120

- Remember to always turn in something. If after you have run out of time and could not meet the requirements, comment out the code that is not working, and add in comments as to what the behavior of the script is. Explain in your write-up what you understand to be happening and you may be awarded partial credit if you were headed in the right direction

Original content from *GEOG 485: GIS Programming and Automation* (https://www.e-education.psu.edu/geog485)
adapted for GISC 2435, Spring 2016