**Shade-a-lator for ArcMap**

Described here are instructions for running shade, a model developed by the Y.D. Chen and the Oregon Department of Environmental Quality (ODEQ) as part of their HeatSource model version 6, from within ArcGis - using TTools (an ArcMap extension, also developed by ODEQ, that estimates input values for shade from GIS coverage). Shade-a-lator and TTools scripts were supplied by Luke Spaete at BSU.

All scripts were tested with ArcMap 10.3.1 running on Windows 7, using Python 2.7.8, *shade\_ver40b04a06.xlsm* (modified by GD to contain some extra macros that allow data transfer to/from the worksheet, now named *shade\_GD.xlsm*) and copies of the TTools python scripts (modified by GD to correct some coding errors and run within ArcMap) and some extra scripts that allow ArcMap to interact with *shade\_GD.xlsm*).

**Installation:**

1. **Copy/have access to ‘scripts’ folder (created by GD) it should contain the following files:**

---------------------------------------------------------------------

*blank.xlsx*

*blankMainMenu.xlsx*

*ExcelToTable.py*

*ExcelToTable.pyc*

*Georges\_Tools.tbx*

*get-pip.py*

*output.xlsx*

*output2.xlsx*

*shade\_GD.xlsm*

*Step1\_SegmentStream.py*

*Step2\_MeasureChannelWidth.py*

*Step3\_SampleElevationGradient\_Array.py*

*Step4\_MeasureTopographicAngles.py*

*Step5\_Sample\_Landcover\_PointMethod\_Array.py*

*Step6\_Interact\_with\_Shade.py*

*Step7\_Import\_shadeData.py*

And the folder (which can be empty):

*data*

------------------------------------------------------------------------------

Beware, if any of the following files/folders are missing the scripts may not run correctly.

**python scripts named *Step\*.py*:** These are the modified TTools code that extract geographical data from ArcMap, create nodes, transects and land cover estimates to input into shade - and then run the shade model (by activating macros within *shade.xlsm*) collect the data outputted by shade and update the nodes with estimated shade values.

***shade\_GD.xlsm*:** This is where the shade model is run. Controlled by macros.

**other *\*.xlsm* files:** Templates used by *Step\*.py* scripts to facilitate movement of data to and from *shade\_GD.xlsm* and ArcMap.

**data:** file structure used for temporary storage of data files

1. **Ensure the Python installation used by ArcMap contains the necessary packages.**

ArcMap 10.3 uses python 2.7 by default. This may well be a separate Python installation to any others installed on your computer. ArcMap’s Python installation is usually found at: C:\Python27\ArcGIS10.3 (for version 10.3).

The easiest way to install libraries into ArcMap’s Python packages is to use pip (a python package manager). First you need to install pip and a C++ compiler (if it is not already installed), so:

1. If necessary, download and install the Windows SDK from Microsoft to gain a compiler (<https://www.microsoft.com/en-us/download/details.aspx?id=8279>)
2. *get-pip.py* should already be in the scripts folder described above. If not, download it to your desktop (or wherever suits you) from <https://bootstrap.pypa.io/get-pip.py>
3. Next, open up the command prompt and navigate to the directory where ArcMap’s Python executable file is located.  
     
   > cd C:\Python27\ArcGIS10.3
4. Finally, directly access the Python 2.7 64-bit executable to run *get-pip.py* and install pip on this version, e.g. (if *get-pip.py* is on your desktop):  
     
   > python.exe C:\Users\<userlogin>\Desktop\get-pip.py

Now you should be able to use pip to install Python packages into ArcMap’s Python library. If you have more than one ArcMap Python installation (i.e. 32 bit and 64 bit versions of Python 2.7 and/or Python 3) you should repeat the process for each installation to make pip universally available.

Most of the packages needed for shade-a-lator should already be installed. However, the following (openpyxl and pypiwin32) need to be installed by running pip from the command window as follows:

1. > cd C:\Python27\ArcGIS10.3\Scripts
2. > pip.exe install openpyxl pypiwin32

If any other packages are missing (you will be informed of their absence by error messages when running the scripts) use pip to install them as above (pip.exe install <package name>)

1. **Make sure Excel is installed on your PC and that macros are enabled.**

You might have to activate developer mode within Excel…

1. **Add Georges\_Tools.tbx to ArcMap**

They can be found in the scripts folder.

1. **Now you should be able to run Shade from within ArcMap.**

As it is setup at the moment, the required input data is:

1. Centerline feature class (OBJECTID, Shape, Shape\_Length, Id)
2. Right bank feature class (OBJECTID, Shape, Shape\_Length, LEFT\_FID, RIGHT\_FID)
3. Left bank feature class (OBJECTID, Shape, Shape\_Length, LEFT\_FID, RIGHT\_FID)
4. Elevation raster (e.g. LidarDEM)

Run each Script in order (Script1, Script2... etc)

The scripts will generate extra feature classes (*nodes\_fc, topo\_fc, lc\_point\_fc* andfinally *shade\_fc*) as they run. The toolbox default settings should point to the correct location for all of these, you just need to point the scripts to the input data listed above. So far I have only tested the scripts with the default settings in the toolbox - but most of them should be able to be changed to whatever values you desire - try them and see…

Once Script7 is run the *shade\_fc* class should be created with the following fields: FID, Shape, STREAM\_ID, ELEVATION, LATITUDE, LONGITUDE, SHADE.

1. **Optional - PyScripter installation**

If you wish to modify the scripts you will most likely wish to install an IDE running Python 2.7. I found that the easiest way to do this was to install PyScripter. I downloaded the installer for the 32-bit version from SourceForge (<https://sourceforge.net/projects/pyscripter/>) and ran it with the suggested install file location *etc*. It should automatically link to the python 2.7 library used by ArcMap.

**Resources:**

Chen, Y.D. (1996). Hydrologic and water quality modeling for aquatic ecosystem protection and restoration in forest watersheds: a case study of stream temperature in the Upper Grande Ronde River, Oregon. PhD dissertation. University of Georgia. Athens, GA.

Chen, Y.D., Carsel, R.F., McCutcheon, S.C., and Nutter, W.L. (1998). Stream temperature simulation of forested riparian areas: I. watershed-scale model development. Journal of Environmental Engineering. April 1998. pp 304-315.

Chen, Y.D., Carsel, R.F., McCutcheon, S.C., and Nutter, W.L. (1998). Stream temperature simulation of forested riparian areas: II. model application. Journal of Environmental Engineering. April 1998. pp 316-328.

Boyd, M., and Kasper, B. 2003. Analytical methods for dynamic open channel heat and mass transfer: Methodology for heat source model Version 7.0.

The ODEQ method was developed by the Oregon Department of Environmental Quality (Heatsource model version 6). Documentation of ODEQ's HeatSource model is available at www.deq.state.or.us/wq/TMDLs/TMDLs.htm and [www.heatsource.info](http://www.heatsource.info)

For more information about *Shade.xls*, contact Greg Pelletier at the Department of Ecology (gpel461@ecy.wa.gov).

**Github repository for TTools:**

https://github.com/rmichie/TTools