# Post-Processed Outputs – MM’s CONUS1 paper

All of MMs CONUS1 outputs were processed using fortran scripts that can be found on /glade/p/univ/ucsm0002/CONUS\_modern/CONUS.WY2003/scripts

These consist of **daily**, **monthly**, and **yearly** averages of the following variables:

*ParFlow Averages:*

* Flow
* Soil moisture (sm)
* Water table depth (WTd)
* Total Subsurface Storage (SUBstorage)
* GW storage (GWstor)
* Soil moisture storage (SMstor)

*Storage sums:*

* Surface Water Storage (surf\_wat)
  + surf\_wat(i,j) = surf\_wat(i,j) + max(press(i,j,5),0.0d0)/ 24.0d0 !avg
* Total water storage (tws) 🡪 subsurface + surface + SWE storages

*CLM Averages:*

* Latent heat (LH) – CLM out layer 1 [W/m^2]
* Sensible heat flux (SH) – CLM out layer 3 [W/m^2]
* ground evaporation without condensation (qflx\_grnd) – CLM out layer 6 [mm/s]
* Vegetation transpiration (qflx\_trans) – CLM out layer 9 [mm/s]
* Snow water equivalent (SWE) – CLM out layer 11 [mm]
* Ground temperature (Tgrnd) – CLM out layer 12 [K] skin temp
* Soil temperature (Tsoil) – CLM out layer 14 [K] @5cm
* Vapor Pressure Deficit (FluxNET)
  + Calculated from NLDAS Air Temp, Air Pressure, and specific humidity

# Scripts To Write

**PFCLM Scripts:**

*PF\_Averages.py*

* Flow
* Soil moisture (sm)
* Water table depth (WTd)

*CLM\_Averages.py*

* Latent heat (LH) – CLM out layer 1 [W/m^2]
* Sensible heat flux (SH) – CLM out layer 3 [W/m^2]
* ground evaporation without condensation (qflx\_grnd) – CLM out layer 6 [mm/s]
* Vegetation transpiration (qflx\_trans) – CLM out layer 9 [mm/s]
* Snow water equivalent (SWE) – CLM out layer 11 [mm]
* Ground temperature (Tgrnd) – CLM out layer 12 [K] skin temp
* Soil temperature (Tsoil) – CLM out layer 14 [K] @5cm

*TotalWaterStorage.py*

* Surface water storage
* Total water storage

*SubsurfaceStorage.py*

* Total Subsurface Storage (SUBstorage)
* GW storage (GWstor)
* Soil moisture storage (SMstor)

**Forcing Scripts:**

THINGS THAT DON’T REQUIRE PF RUNS COMPLETE:

* FluxNET
  + comparison of vapor pressure deficit, air temp, and wind speed
  + MM’s scripts: /Validation/FLUXNET
    - Map\_FluxNET.R
    - Organize\_Data.R
* GHCND
  + Temperature and precipitation
  + MM’s scripts: /glade/p/univ/ucsm0002/CONUS\_modern/Analysis\_Validation/Validation/GHCND\_MetStations
  + My Python scripts: /glade/work/tijerina/CONUS2\_work/Validation/GHCND
* SNOTEL
  + Temperature and precipitation

# From Jackson 5/14/22

Subsurface in General

1. The subsurface is 10 layers thick, a twofold increase from CONUS 1.0. The depths can be seen in a figure on slides 7 & 8
2. The top four layers are soil and the bottom six are geology
3. We make use of indicators, which act as functional soil types. A value for permeability is assigned by indicator. These values are the same as CONUS 1.0
4. We place a vertical flow barrier between cells at a depth defined by Shangguan’s unconsolidated thickness dataset. The vertical flow barrier only reduces flow to 0.1% instead of completely preventing.
5. We tested subsurfaces and manning’s roughnesses in the Upper Colorado, the Little Washita, and Jun’s domain ICOM domain

Soil

1. We use SSURGO data inside US boundaries and GSDE soil data outside of the US.
2. We take percent clay, sand, and silt to define 13 (only 12 were present in the US) soil types following the USDA soil texture classifications. A list of the soil types and their properties are shown on slide 7.
3. In addition to permeability, soil parameters such as porosity and Van Genuchten parameters (soil water retention) vary by soil indicator.
4. SSURGO soil color data was used to define soil color for land surface modeling (clm\_drv\_vegm.dat). To extrapolate outside of the US, averages were found for each indicator, and color was assigned by indicator (this was only done outside of the US, the RGB values from SSURGO were used inside).
5. The percent sand, clay, and from the combined soil data was used to as an input for CLM as well (clm\_drv\_vegm.dat).
6. Conveniently, both datasets were provided at the 4 depths of our interest if I remember correctly.

Geology

1. We use GLHYMPS 1.0 geology data mapped to our indicator scheme
2. We include two bedrock geologies of our own design
3. We perform e-folding to account for the exponential decay of hydraulic conductivity with depth. (Hoang wrote the e-folding part of our script, so I’m not entirely sure what goes on under the hood)
4. Values are recycled from CONUS 1.0, which are discussed by Condon and Maxwell 2014
5. We tested a whole shitload of these, but this was the best performing
6. I honestly forget if we settled on 0.1 or no anisotropy in the vertical direction, but we could check the script.

Surface Configuration

Landcover

1. Jun was in charge of this. I do know that tested manning’s values. I think we considered lowering them and then went back to the originals.
2. One important feature is that manning’s roughness is mapped by landcover type, and we lowered them in stream cells. The higher the stream order, the lower the n.

Slopes

This was done while I was still trying to get a bearing on running ParFlow, so I have even less insightful information.

1. I do know that everything was processed using priority flow and Jun published a paper describing the methods.
2. Jun spent some time laboring over comparisons to USGS drainage areas and (I believe) manually fixing large problem areas.

Variable manning’s – based on landcover

Soil data – comes from dataset, normalized for CONUS in indicator file

Geology – GLYMPHS 1.0 (performed better than 2.0, there was massive clay around Miss river) 2.0 mapped unconsolidated layers, which we already did pretty well with soil.

Landcover – updated? Upscaled from 30m

New datasets

* Topography, priority flow processing D4 connectedness, sink exist
  + Sinks, lakes, and coastlines
* New landcover, variable mannings
* Depth to bedrock – separate unconsolidated layers from deeper gw systems

Extensive testing

* Range of climates
* >50 different combinations of parameter combinations in 3 domains