# The Future of the AMWG diagnostics package?

*Cécile Hannay, Brian Medeiros, Jesse Nusbaumer*
*with contributions of Dani Coleman, Matt Long, and AMP/CGD.*

March 11, 2020

This material is based upon work supported by the National Center for Atmospheric Research, which is a major facility sponsored by the National Science Foundation under Cooperative Agreement No. 1852977.

# Motivation and Outline

## Motivation

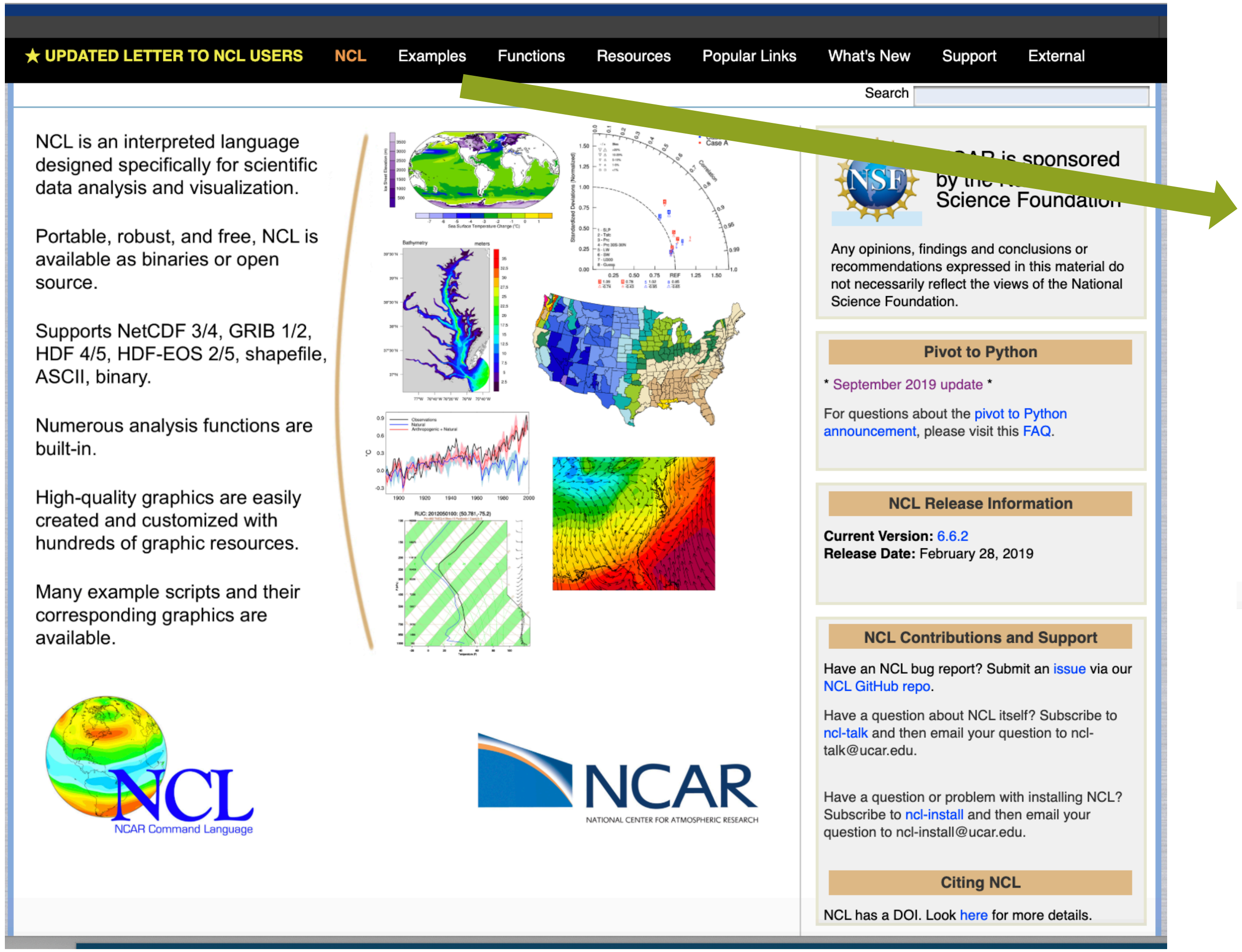
- NCL is going away
- NCAR is transitioning to Python for diagnostics
- How is going to affect the AMWG diagnostics ?

## Outline

- What are we using now ?
- What are our needs for the future ?
- How does it fit in the NCAR vision for diagnostics ?
  (Other packages in development)

## What are we using now ?

# The NCAR Command Language (NCL)



★ UPDATED LETTER TO NCL USERS | NCL | Examples | Functions | Resources | Popular Links | What's New | Support | External

Search

NCL is an interpreted language designed specifically for scientific data analysis and visualization.

Portable, robust, and free, NCL is available as binaries or open source.

Supports NetCDF 3/4, GRIB 1/2, HDF 4/5, HDF-EOS 2/5, shapefile, ASCII, binary.

Numerous analysis functions are built-in.

High-quality graphics are easily created and customized with hundreds of graphic resources.

Many example scripts and their corresponding graphics are available.

NCL
NCAR Command Language

NCAR
NATIONAL CENTER FOR ATMOSPHERIC RESEARCH

NCAR is sponsored by the National Science Foundation

Any opinions, findings and conclusions or recommendations expressed in this material do not necessarily reflect the views of the National Science Foundation.

**Pivot to Python**

* September 2019 update *

For questions about the pivot to Python announcement, please visit this FAQ.

**NCL Release Information**

**Current Version:** 6.6.2
**Release Date:** February 28, 2019

**NCL Contributions and Support**

Have an NCL bug report? Submit an issue via our NCL GitHub repo.

Have a question about NCL itself? Subscribe to ncl-talk and then email your question to ncl-talk@ucar.edu.

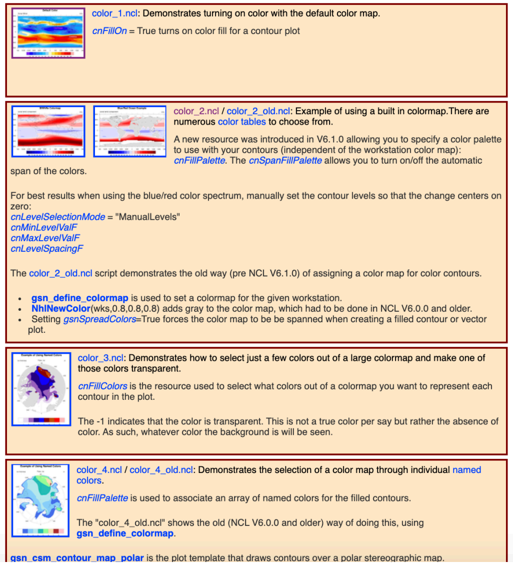
Have a question or problem with installing NCL? Subscribe to ncl-install and then email your question to ncl-install@ucar.edu.

**Citing NCL**

NCL has a DOI. Look here for more details.

https://www.ncl.ucar.edu/

## Look at the examples



color_1.ncl: Demonstrates turning on color with the default color map.

*cnFillOn* = True turns on color fill for a contour plot

color_2.ncl / color_2_old.ncl: Example of using a built in colormap.There are numerous color tables to choose from.

A new resource was introduced in V6.1.0 allowing you to specify a color palette to use with your contours (independent of the workstation color map): *cnFillPalette*. The *cnSpanFillPalette* allows you to turn on/off the automatic span of the colors.

For best results when using the blue/red color spectrum, manually set the contour levels so that the change centers on zero:
*cnLevelSelectionMode* = "ManualLevels"
*cnMinLevelValF*
*cnMaxLevelValF*
*cnLevelSpacingF*

The color_2_old.ncl script demonstrates the old way (pre NCL V6.1.0) of assigning a color map for color contours.

- **gsn_define_colormap** is used to set a colormap for the given workstation.
- **NhlNewColor**(wks,0.8,0.8,0.8) adds gray to the color map, which had to be done in NCL V6.0.0 and older.
- Setting *gsnSpreadColors*=True forces the color map to be be spanned when creating a filled contour or vector plot.

color_3.ncl: Demonstrates how to select just a few colors out of a large colormap and make one of those colors transparent.

*cnFillColors* is the resource used to select what colors out of a colormap you want to represent each contour in the plot.

The -1 indicates that the color is transparent. This is not a true color per say but rather the absence of color. As such, whatever color the background is will be seen.

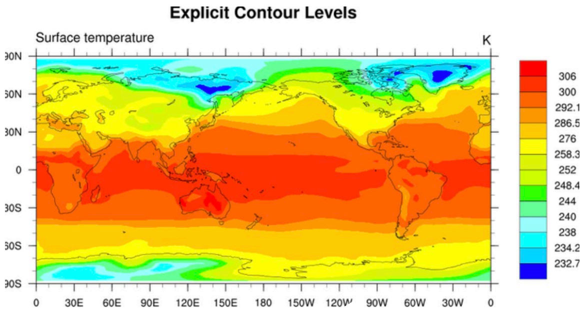
color_4.ncl / color_4_old.ncl: Demonstrates the selection of a color map through individual named colors.

*cnFillPalette* is used to associate an array of named colors for the filled contours.

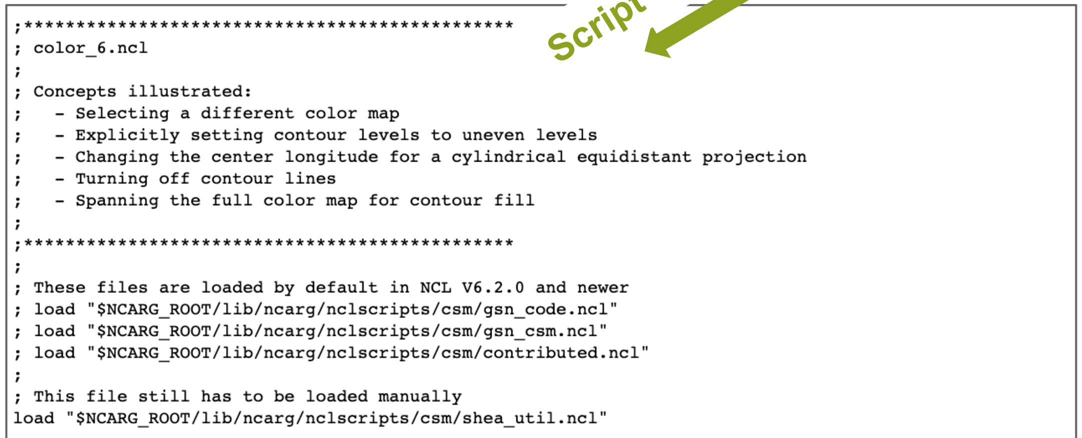
The "color_4_old.ncl" shows the old (NCL V6.0.0 and older) way of doing this, using **gsn_define_colormap**.

**gsn_csm_contour_map_polar** is the plot template that draws contours over a polar stereographic map.

Find a plot I like



Script to do that plot



```
;*************************************************
; color_6.ncl
;
; Concepts illustrated:
;   - Selecting a different color map
;   - Explicitly setting contour levels to uneven levels
;   - Changing the center longitude for a cylindrical equidistant projection
;   - Turning off contour lines
;   - Spanning the full color map for contour fill
;
;*************************************************
;
; These files are loaded by default in NCL V6.2.0 and newer
; load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/gsn_code.ncl"
; load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/gsn_csm.ncl"
; load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/contributed.ncl"
;
; This file still has to be loaded manually
load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/shea_util.ncl"
```

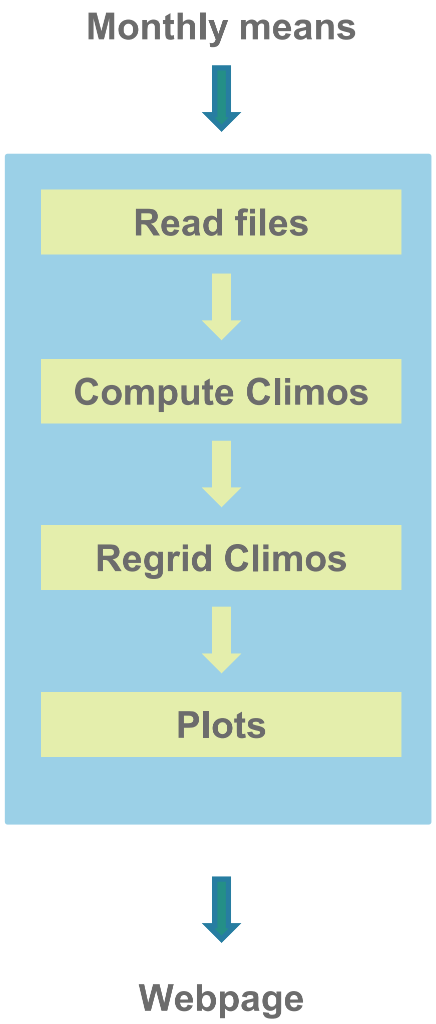
**GeoCAT (https://geocat.ucar.edu/) python tools related to NCL**

**NCL is retiring**

# The AMWG Diagnostics package

cshell script calling

- NCO
- NCL
- ImageMagic


Monthly means

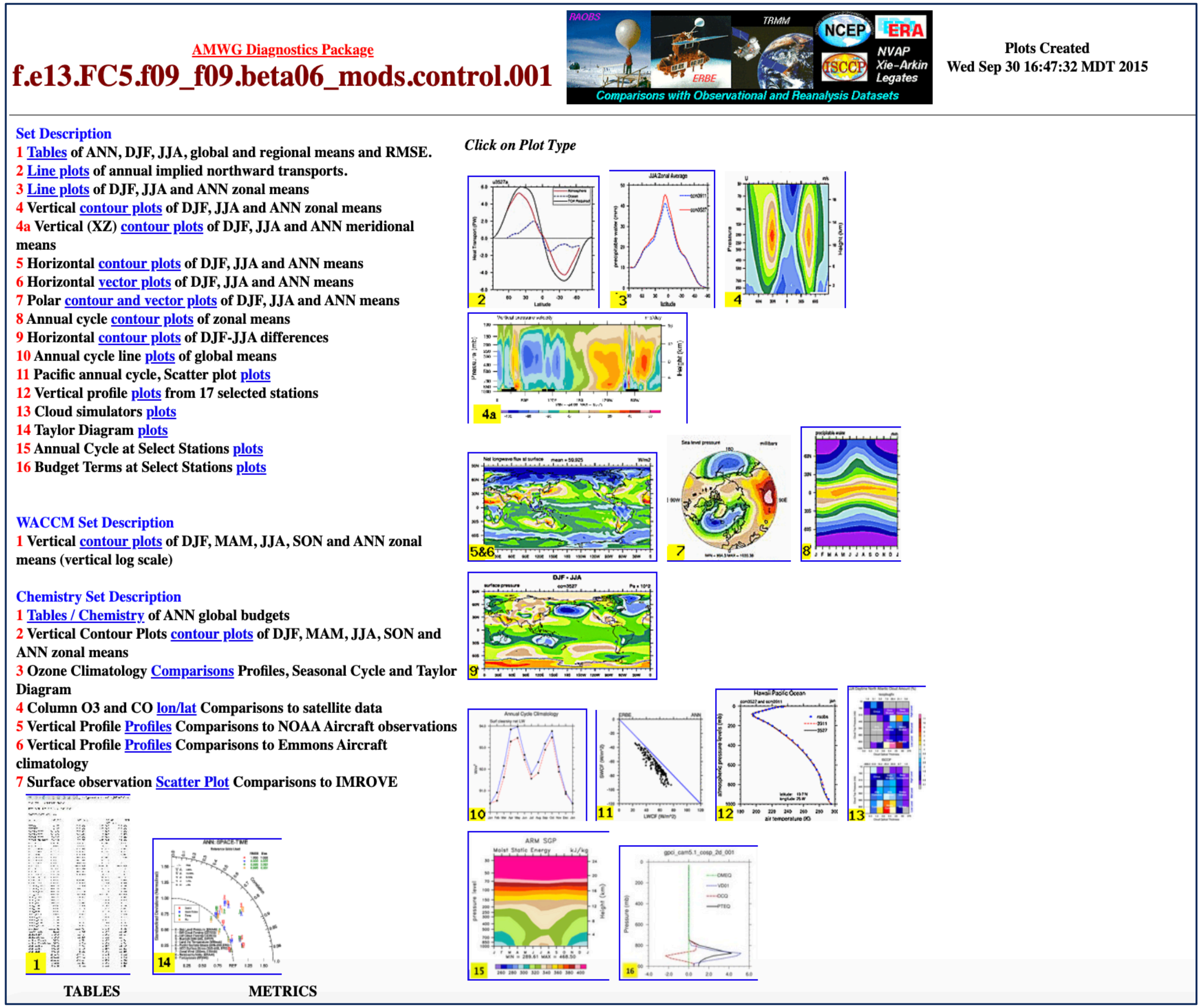
Read files

Compute Climos

Regrid Climos

Plots

Webpage


AMWG Diagnostics Package

f.e13.FC5.f09_f09.beta06_mods.control.001

Plots Created
Wed Sep 30 16:47:32 MDT 2015

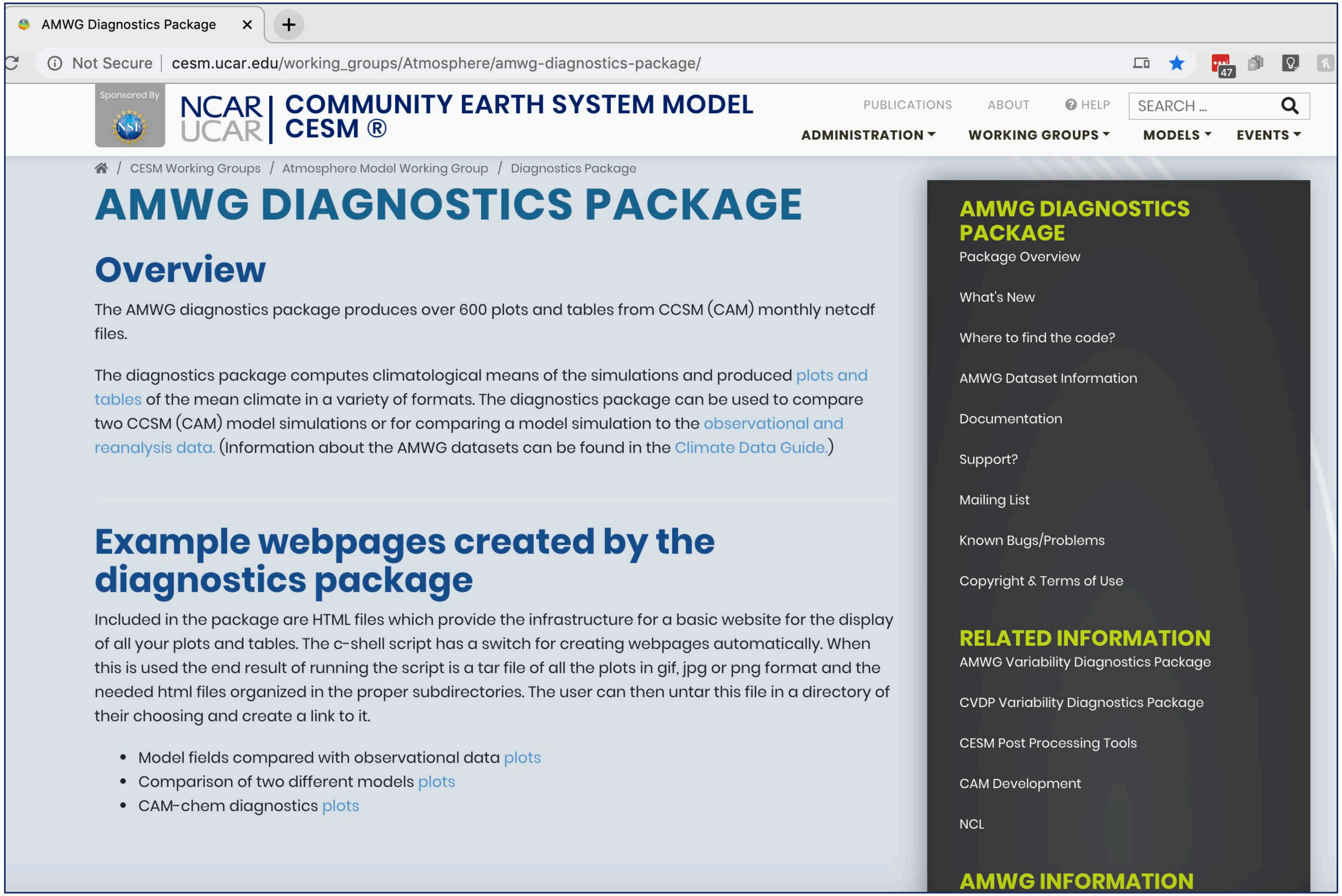
Set Description
1 Tables of ANN, DJF, JJA, global and regional means and RMSE.
2 Line plots of annual implied northward transports.
3 Line plots of DJF, JJA and ANN zonal means
4 Vertical contour plots of DJF, JJA and ANN zonal means
4a Vertical (XZ) contour plots of DJF, JJA and ANN meridional means
5 Horizontal contour plots of DJF, JJA and ANN means
6 Horizontal vector plots of DJF, JJA and ANN means
7 Polar contour and vector plots of DJF, JJA and ANN means
8 Annual cycle contour plots of zonal means
9 Horizontal contour plots of DJF-JJA differences
10 Annual cycle line plots of global means
11 Pacific annual cycle, Scatter plot plots
12 Vertical profile plots from 17 selected stations
13 Cloud simulators plots
14 Taylor Diagram plots
15 Annual Cycle at Select Stations plots
16 Budget Terms at Select Stations plots

WACCM Set Description
1 Vertical contour plots of DJF, MAM, JJA, SON and ANN zonal means (vertical log scale)

Chemistry Set Description
1 Tables / Chemistry of ANN global budgets
2 Vertical Contour Plots contour plots of DJF, MAM, JJA, SON and ANN zonal means
3 Ozone Climatology Comparisons Profiles, Seasonal Cycle and Taylor Diagram
4 Column O3 and CO lon/lat Comparisons to satellite data
5 Vertical Profile Profiles Comparisons to NOAA Aircraft observations
6 Vertical Profile Profiles Comparisons to Emmons Aircraft climatology
7 Surface observation Scatter Plot Comparisons to IMROVE

*Click on Plot Type*

# The AMWG Diagnostics package


AMWG Diagnostics Package
Not Secure
cesm.ucar.edu/working_groups/Atmosphere/amwg-diagnostics-package/
Sponsored By
NCAR UCAR
COMMUNITY EARTH SYSTEM MODEL CESM ®
PUBLICATIONS
ABOUT
HELP
SEARCH ...
ADMINISTRATION
WORKING GROUPS
MODELS
EVENTS
CESM Working Groups / Atmosphere Model Working Group / Diagnostics Package

## AMWG DIAGNOSTICS PACKAGE

### Overview

The AMWG diagnostics package produces over 600 plots and tables from CCSM (CAM) monthly netcdf files.

The diagnostics package computes climatological means of the simulations and produced plots and tables of the mean climate in a variety of formats. The diagnostics package can be used to compare two CCSM (CAM) model simulations or for comparing a model simulation to the observational and reanalysis data. (Information about the AMWG datasets can be found in the Climate Data Guide.)

### Example webpages created by the diagnostics package

Included in the package are HTML files which provide the infrastructure for a basic website for the display of all your plots and tables. The c-shell script has a switch for creating webpages automatically. When this is used the end result of running the script is a tar file of all the plots in gif, jpg or png format and the needed html files organized in the proper subdirectories. The user can then untar this file in a directory of their choosing and create a link to it.

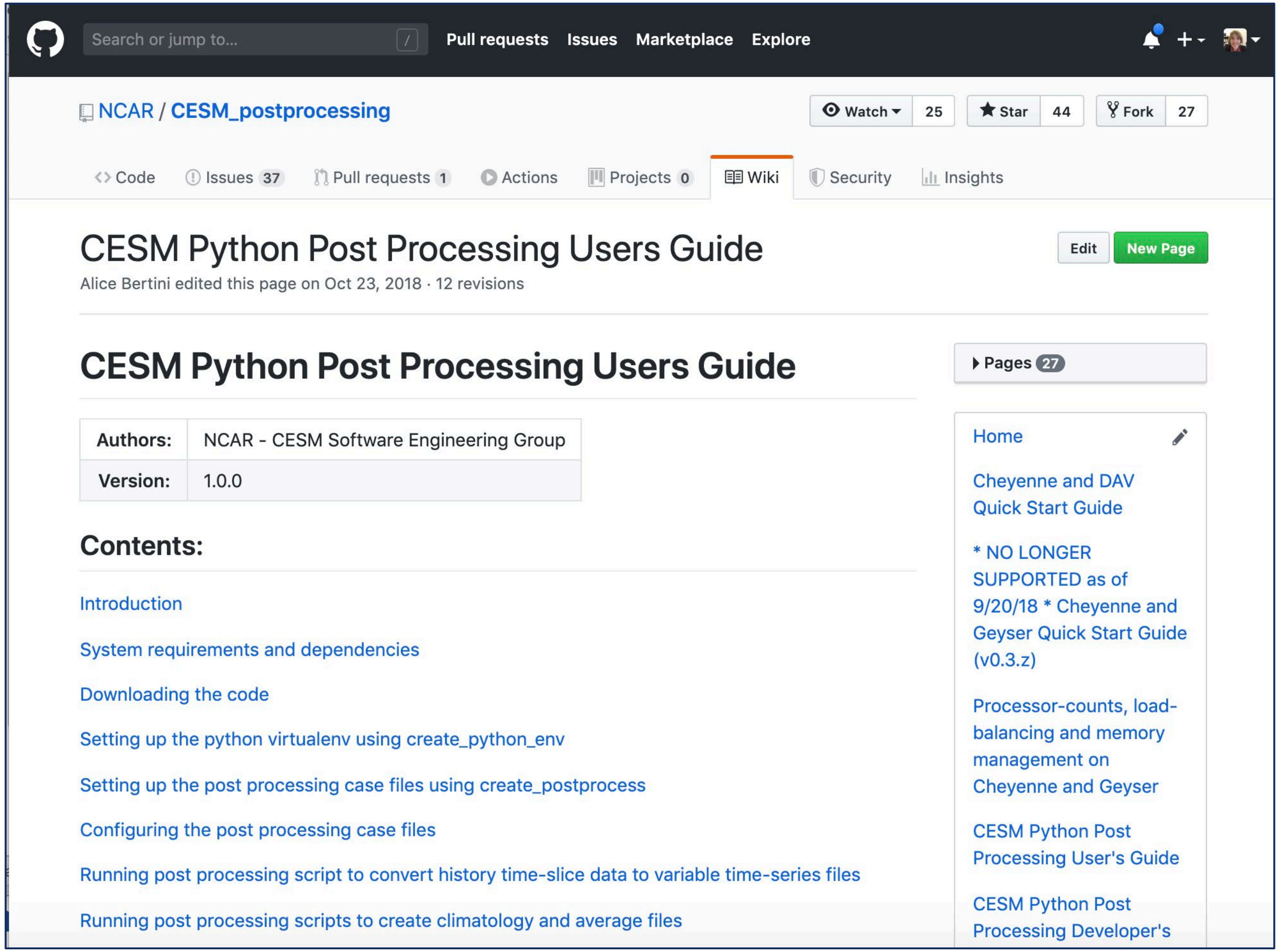
- Model fields compared with observational data plots
- Comparison of two different models plots
- CAM-chem diagnostics plots

http://www.cesm.ucar.edu/working_groups/Atmosphere/amwg-diagnostics-package/

Well documented
Easy to use
Portable
Easy to install anywhere

20-year old
Very 2000$^{ish}$ code
Driver = cshell
No more supported
NCL is going away

# The CESM postprocessing package



CESM Python Post Processing Users Guide

Alice Bertini edited this page on Oct 23, 2018 · 12 revisions

## CESM Python Post Processing Users Guide

| | |
|---|---|
| **Authors:** | NCAR - CESM Software Engineering Group |
| **Version:** | 1.0.0 |

**Contents:**

- Introduction
- System requirements and dependencies
- Downloading the code
- Setting up the python virtualenv using create_python_env
- Setting up the post processing case files using create_postprocess
- Configuring the post processing case files
- Running post processing script to convert history time-slice data to variable time-series files
- Running post processing scripts to create climatology and average files

Home

Cheyenne and DAV Quick Start Guide

* NO LONGER SUPPORTED as of 9/20/18 * Cheyenne and Geyser Quick Start Guide (v0.3.z)

Processor-counts, load-balancing and memory management on Cheyenne and Geyser

CESM Python Post Processing User's Guide

CESM Python Post Processing Developer's

👍

- Common framework that produces diags for all components
- Produced CMIP6 diags automatically
- Well documented

👎

- Complex framework to ultImately run NCL scripts
- Very complex (for scientists)
- Not portable (only run on Cheyenne)
- Dependency on outdated packages (python2, basemap,,…)
- Not more support

https://github.com/NCAR/CESM_postprocessing/wiki/CESM-Python-Post-Processing-Users-Guide

# What are our needs for the future ?

# What are our needs and where do we want to go from here ?

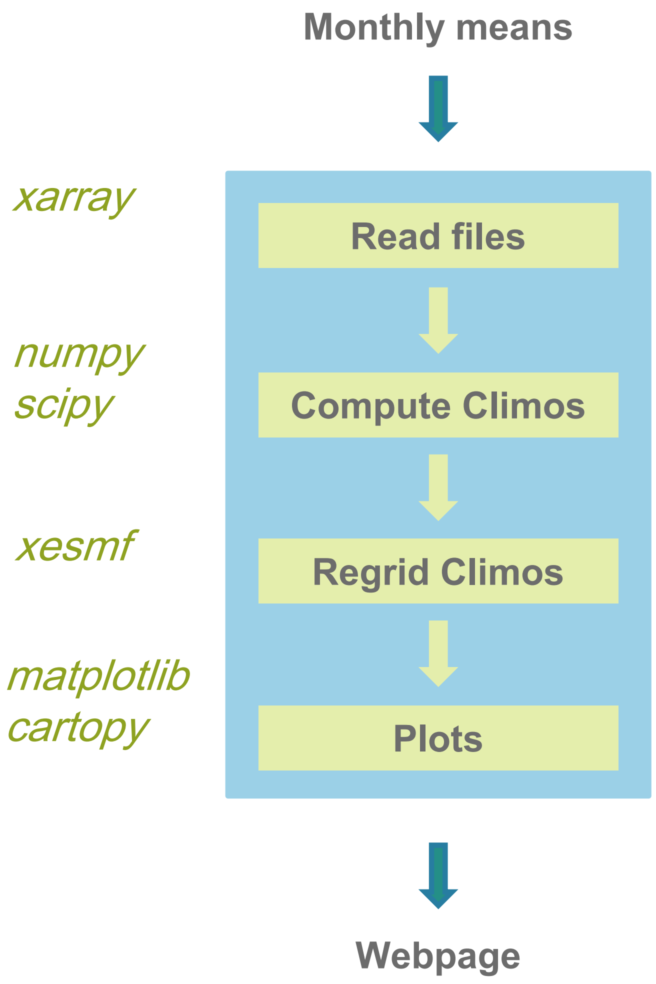
## Package to have a quick look at simulations

In absence of support + NCL going away:
- Simple framework python based
- Scientists can understand and modify
- Portability (can run anywhere)
- Small numbers of python packages

## Bank of scripts to replace NCL examples

# What are our needs and where do we want to go from here ?

## Package to have a quick look at simulations



*yaml*

How to run:

```
run_diag < namelist
```

namelist

- casename
- climo year

Code would live on gitub

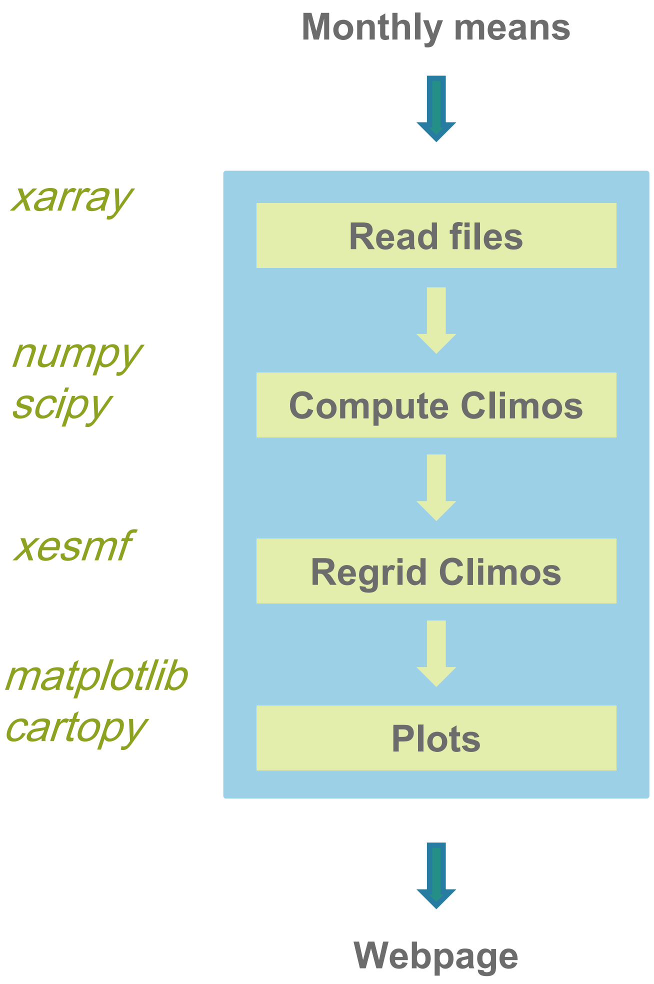
https://github.com/NCAR

Contributions would be through review process

## Bank of scripts to replace NCL examples

# What are our needs and where do we want to go from here ?

## Package to have a quick look at simulations



Monthly means

*xarray*

Read files

*numpy*
*scipy*

Compute Climos

*xesmf*

Regrid Climos

*matplotlib*
*cartopy*

Plots

Webpage

*yaml*

How to run:

```
run_diag < namelist
```

namelist

- casename
- climo year

Code would live on gitub

https://github.com/NCAR

Contributions would be through review process

## Bank of scripts to replace NCL examples

- Collection of python scripts
- Jupyter Notebooks

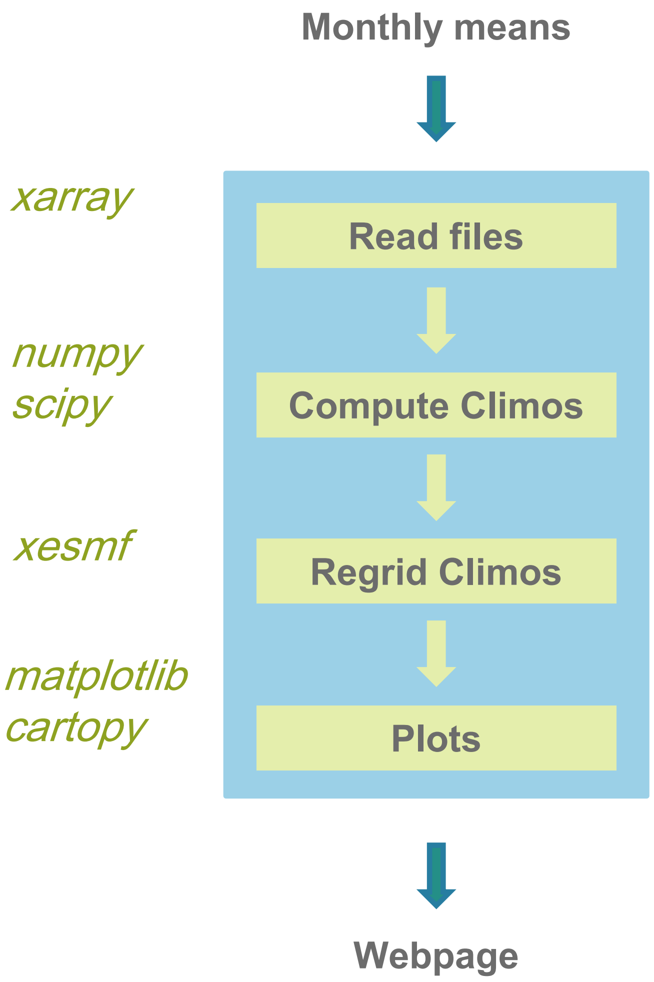
An example of what it could be

https://github.com/brianpm/hacknostics

- Some atmospheric diagnostics, focused on analysis of CESM.
- The repo is organized into Notebooks, utilities (util), and notes (docs).

# What are our needs and where do we want to go from here ?

## Package to have a quick look at simulations



Monthly means

*xarray*

Read files

*numpy*
*scipy*

Compute Climos

*xesmf*

Regrid Climos

*matplotlib*
*cartopy*

Plots

Webpage

*yaml*

How to run:

```
run_diag < namelist
```

namelist
- casename
- climo year

Code would live on gitub
https://github.com/NCAR

Contributions would be through review process

## Bank of scripts to replace NCL examples

- Collection of python scripts
- Jupyter Notebooks

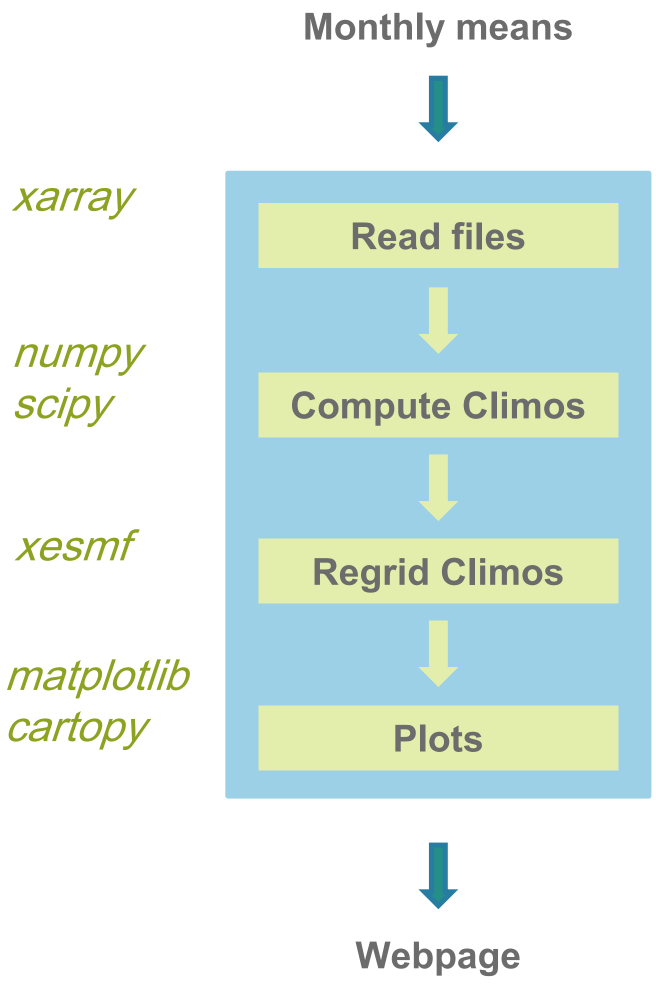
Code would live on gitub
https://github.com/NCAR

Contributions from the community welcome

Could be reorganized (like NCL Webpage)

# What are our needs and where do we want to go from here ?

## Package to have a quick look at simulations



Monthly means

xarray

Read files

numpy
scipy

Compute Climos

xesmf

Regrid Climos

matplotlib
cartopy

Plots

Webpage

*yaml*

How to run:

```
run_diag < namelist
```

namelist
- casename
- climo year

Code would live on gitub
[https://github.com/NCAR](https://github.com/NCAR)

Contributions would be through review process

## Bank of scripts to replace NCL examples

- Collection of python scripts
- Jupyter Notebooks

Code would live on gitub
[https://github.com/NCAR](https://github.com/NCAR)

Contributions from the community welcome

Could be reorganized (like NCL Webpage)

**Deadline for first implementation: April 15, 2020**

# How does this fit in the bigger picture ?

# Pangeo : Core tools

PANGEO

jupyter

**A thin client** Interactive computing in the universal app: a browser; produce and publish “computational narratives.”

xarray

**Python netCDF data model** Multi-dimensional datasets w/ coordinates; coordinate-aware selection and dim-reduction methods.

DASK

**Parallel & “out-of-core” computation** Data bigger than memory; parallelized execution on a distributed resource “under the hood”!

Courtesy Matt Long

## Vision: An interactive numerical laboratory for Earth system science

- Seamless integration of routine model evaluation and cutting-edge research
- Enable reproducible science
- Enable novel means of data interactivity and visualization
- Component models are not necessarily a natural organizing principle
    - Be as model agnostic as much as possible
- Scalable
    - Big Data
    - New applications, communities, etc.
- Fluid integration of observations and models
- Community-developed and open-source

Courtesy Matt Long

# Vision: An interactive numerical laboratory for Earth system science

- **Analysis elements:** modular scripts or Notebooks that perform computation and support visualization
- **Data APIs** : abstract data access through APIs
- **Operators** : perform dimensions reductions, compute derived quantities, etc.
- **Visualization** : static plots, web apps
- **Workflow** : Automate, schedule, interact

- Build library of `analysis_element` prototypes
  - Aim for best practices (modularity!), but focus on scientific objectives and achieving basic function first
  - Coordinate sharing and standardization
  - Communicate!
- Reassess and coalesce as library grows

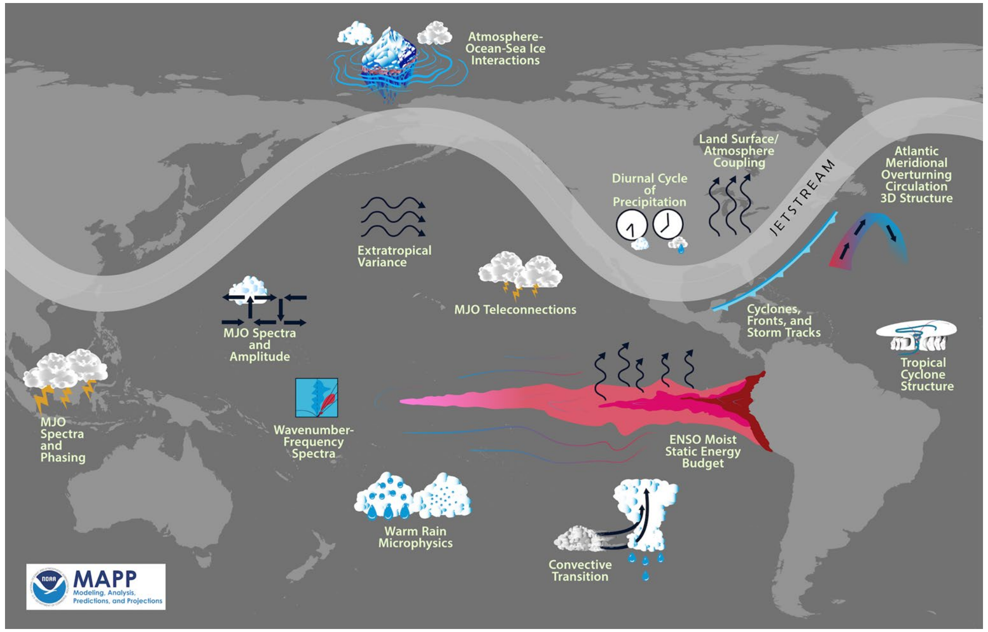
Courtesy Matt Long

Matt will have a talk at OMWG workshop

# Model Diagnostics Task Force framework

An open framework to run process-oriented diagnostics from research teams to inform model development.

- Focused on specific phenomena and process-level understanding
- Available for anyone to use the software package (python and ncl) to apply to a model simulation (cmip6 variables). Currently 7 packages. More coming including multi-model capability.


Atmosphere-Ocean-Sea Ice Interactions
Land Surface/Atmosphere Coupling
Diurnal Cycle of Precipitation
JETSTREAM
Atlantic Meridional Overturning Circulation 3D Structure
Extratropical Variance
MJO Teleconnections
Cyclones, Fronts, and Storm Tracks
MJO Spectra and Amplitude
Tropical Cyclone Structure
MJO Spectra and Phasing
Wavenumber-Frequency Spectra
ENSO Moist Static Energy Budget
Warm Rain Microphysics
Convective Transition
MAPP
Modeling, Analysis, Predictions, and Projections

Diagnostics include:
-Cloud microphysical processes
-Tropical & extratropical cyclones
-ENSO teleconnections
-MJO moisture, convection, & radiative processes
-Precipitation diurnal cycle
-AMOC, Pacific sea level variability
-Arctic sea ice
-Lake effect processes
-North American Monsoon
-Radiative forcing & cloud-circulation feedbacks
-Temperature & precip. extremes
more!...

www.cesm.ucar.edu/working_groups/Atmosphere/mdtf-diagnostics-package/

Maloney et al. (2019, *BAMS*)

Courtesy Dani Coleman

# Summary

- Package to have a quick look at diagnostics
  - simple, portable, small number of python packages
- Bank of python scripts and Jupyter Notebooks
  - Community contributions encouraged
- Code would live on gitub
  - https://github.com/NCAR
- Timeline
  - First implementation: April 15, 2020
- Inline with the NCAR vision for future diagnostics